ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MANUSCRIPT-BASED THESIS PRESENTED TO
ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph. D.

BY
CHRISTOPHE PAGANO

ADAPTIVE CLASSIFIER ENSEMBLES FOR FACE RECOGNITION IN VIDEO-SURVEILLANCE

MONTREAL, OCTOBER 8, 2015

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS:

Dr. Éric Granger, Thesis Supervisor
Département de génie de la production automatisée, École de Technologie Supérieure

Dr. Robert Sabourin, Co-supervisor
Département de génie de la production automatisée, École de Technologie Supérieure

Dr. Stéphane Coulombe, President of the Board of Examiners
Département de génie logiciel et des TI, École de Technologie Supérieure

Dr. Luc Duong, Member of the Jury
Département de génie logiciel et des TI, École de Technologie Supérieure

Dr. Bernadette Dorizzi, External Examiner
Département électronique et physique, Télécom SudParis, France

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON SEPTEMBER 25, 2015

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest gratitude to my advisors Éric Granger and Robert Sabourin for their personal support and scientific insights. A PhD is a long and bumpy road, and I'm very grateful for the many times they helped me get back on my feet to keep going in the right direction. I am also very grateful for the financial support, that helped me to focus on my research with an unburdened mind.

I would also like to thank all my friends and coworkers from the LIVIA, for helping me with their great ideas, making me feel at home and patiently listening to all my rants when I needed to vent. A special thanks to Jean-François for sharing great music and for all the wine and cheese we gobbled at the lab, to Olaf and his delicious concoctions, to Paulo and Rafael for the many passionate discussions about video games and music, and to Idrissa, whose astonishing reliability is only equalled by his unfailing willingness to help others.

A warm and heartfelt thank you to my family, who, despite being on the other side of the Atlantic Ocean, never left my side. Thank you Mom and Dad for your unconditional love and boundless support, and for making me the way I am, I wouldn't want it any other way ! Thank you Sebastien for your discreet, fraternal presence, and your steadfast enthusiasm. Thank you Alice for your love and thoughtfulness, and unfailing positivity.

A big thank you to all the gang from Montreal, past and present. You are way too numerous to cite here, and it would be impossible to select only a certain few ! A special thanks to Jean-Baptiste, for enduring these past years of flat sharing with me !

And last but not least, I would like to thank you, Lauranne, from the bottom of my heart. Thank you for giving me the chance to get to know you, for your inexhaustible support, and for always being able to lift my spirits with your amazing positivity. Thank you for being you... I love you.

ENSEMBLES DE CLASSIFICATEURS ADAPTATIFS POUR LA RECONNAISSANCE DE VISAGE EN VIDEO-SURVEILLANCE

CHRISTOPHE PAGANO

RÉSUMÉ

Lors de l'implémentation de systèmes de sécurité tels que la vidéo-surveillance intelligente, l'utilisation d'images de visages présente de nombreux avantages par rapport à d'autres traits biométriques. En particulier, cela permet de détecter d'éventuels individus d'intérêt de manière discrète et non intrusive, ce qui peut être particulièrement avantageux dans des situations comme la détection d'individus sur liste noire, la recherche dans des données archivées ou la ré-identification de visages.

Malgré cela, la reconnaissance de visages reste confrontée à de nombreuses difficultés propres à la vidéo-surveillance. Entre autres, le manque de contrôle sur l'environnement observé implique de nombreuses variations dans les conditions d'éclairage, la résolution de l'image, le flou de mouvement, l'orientation et l'expression des visages. Pour reconnaître des individus, des modèles de visages sont habituellement générés à l'aide d'un nombre limité d'images ou de vidéos de référence collectées lors de sessions d'inscription. Cependant, ces acquisitions ne se déroulant pas nécessairement dans les mêmes conditions d'observation, les données de référence représentent pas toujours la complexité du problème réel. D'autre part, bien qu'il soit possible d'adapter les modèles de visage lorsque de nouvelles données de référence deviennent disponibles, un apprentissage incrémental basé sur des données significativement différentes expose le système à un risque de corruption de connaissances. Enfin, seule une partie de ces connaissances est effectivement pertinente pour la classification d'une image donnée.

Dans cette thèse, un nouveau système est proposé pour la détection automatique d'individus d'intérêt en vidéo-surveillance. Plus particulièrement, celle-ci se concentre sur un scénario centré sur l'utilisateur, où un système de reconnaissance de visages est intégré à un outil d'aide à la décision pour alerter un opérateur lorsqu'un individu d'intérêt est détecté sur des flux vidéo. Un tel système se doit d'être capable d'ajouter ou supprimer des individus d'intérêt durant son fonctionnement, ainsi que de mettre à jour leurs modèles de visage dans le temps avec des nouvelles données de référence. Pour cela, le système proposé se base sur de la détection de changement de concepts pour guider une stratégie d'apprentissage impliquant des ensembles de classificateurs. Chaque individu inscrit dans le système est représenté par un ensemble de classificateurs à deux classes, chacun étant spécialisé dans des conditions d'observation différentes, détectées dans les données de référence. De plus, une nouvelle règle pour la fusion dynamique d'ensembles de classificateurs est proposée, utilisant des modèles de concepts pour estimer la pertinence des classificateurs vis-à-vis de chaque image à classifier. Enfin, les visages sont suivis d'une image à l'autre dans le but de les regrouper en trajectoires, et accumuler les décisions dans le temps.

Au Chapitre 2, la détection de changement de concept est dans un premier temps utilisée pour limiter l'augmentation de complexité d'un système d'appariement de modèles adoptant une stratégie de mise à jour automatique de ses galeries. Une nouvelle approche sensible au contexte est proposée, dans laquelle seules les images de haute confiance capturées dans des conditions d'observation différentes sont utilisées pour mettre à jour les modèles de visage. Des expérimentations ont été conduites avec trois bases de données de visages publiques. Un système d'appariement de modèles standard a été utilisé, combiné avec un module de détection de changement dans les conditions d'illumination. Les résultats montrent que l'approche proposée permet de diminuer la complexité de ces systèmes, tout en maintenant la performance dans le temps.

Au Chapitre 3, un nouveau système adaptatif basé des ensembles de classificateurs est proposé pour la reconnaissance de visages en vidéo-surveillance. Il est composé d'un ensemble de classificateurs incrémentaux pour chaque individu inscrit, et se base sur la détection de changement de concepts pour affiner les modèles de visage lorsque de nouvelles données sont disponibles. Une stratégie hybride est proposée, dans laquelle des classificateurs ne sont ajoutés aux ensembles que lorsqu'un changement abrupt est détecté dans les données de référence. Lors d'un changement graduel, les classificateurs associés sont mis à jour, ce qui permet d'affiner les connaissances propres au concept correspondant. Une implémentation particulière de ce système est proposée, utilisant des ensembles de classificateurs de type Fuzzy-ARTMAP probabilistes, générés et mis à jour à l'aide d'une stratégie basée sur une optimisation par essaims de particules dynamiques, et utilisant la distance de Hellinger entre histogrammes pour détecter des changements. Les simulations réalisées sur la base de donnée de vidéo-surveillance Faces in Action (FIA) montrent que le système proposé permet de maintenir un haut niveau de performance dans le temps, tout en limitant la corruption de connaissance. Il montre des performances de classification supérieure à un système similaire passif (sans détection de changement), ainsi qu'a des systèmes de référence de type kNN probabiliste, et TCM-kNN.

Au Chapitre 4, une évolution du système présenté au Chapitre 3 est proposée, intégrant des mécanismes permettant d'adapter dynamiquement le comportement du système aux conditions d'observation changeantes en mode opérationnel. Une nouvelle règle de fusion basée sur de la pondération dynamique est proposée, assignant à chaque classificateur un poids proportionnel à son niveau de compétence estimé vis-à-vis de chaque image à classifier. De plus, ces compétences sont estimées à l'aide des modèles de concepts utilisés en apprentissage pour la détection de changement, ce qui permet un allègement des ressources nécessaires en mode opérationnel. Une évolution de l'implémentation proposée au Chapitre 3 est présentée, dans laquelle les concepts sont modélisés à l'aide de l'algorithme de partitionnement Fuzzy C-Means, et la fusion de classificateurs réalisée avec une moyenne pondérée. Les simulation expérimentales avec les bases de données de vidéo-surveillance FIA et Chokepoint montrent que la méthode de fusion proposée permet d'obtenir des résultats supérieurs à la méthode de sélection dynamique DS-OLA, tout en utilisant considérablement moins de ressources de calcul. De plus, la méthode proposée montre des performances de classification supérieures aux systèmes de référence de type kNN probabiliste, TCM-kNN et Adaptive Sparse Coding.

Mots clés: Systèmes multi-classificateurs, apprentissage incrémental, détection de changement, sélection et fusion dynamique.

ADAPTIVE CLASSIFIER ENSEMBLES FOR FACE RECOGNITION IN VIDEO-SURVEILLANCE

CHRISTOPHE PAGANO

ABSTRACT

In the past decades, face recognition (FR) has received a growing attention in security applications such as intelligent video surveillance (VS). Embedded in decision support tools, FR allows to detect the presence of individuals of interest in video streams in a discrete and non-intrusive way, which is of a particular interest for applications such as watchlist screening, search and retrieval or face re-identification. However, recognizing faces corresponding to target individuals remains a challenging problem in VS. FR systems are usually presented with videos exhibiting a wide range of variations caused by uncontrolled observation conditions, most notably in illumination condition, image resolution, motion blur, facial pose and expression. To perform recognition, facial models of target individuals are typically designed with a limited number of reference stills or videos captured during an enrollment process, and these variations contribute to a growing divergence between these models and the underlying data distribution. Although facial models can be adapted when new reference videos that may become available over time, incremental learning with faces captured under different conditions remains challenging, as it may lead to knowledge corruption. Furthermore, only a subset of this knowledge may be relevant to classify a given facial capture, and relying on information related to different capture conditions may even deteriorate system performance.

In this thesis, a new framework is proposed for the automatic detection of individuals of interest for VS applications. A human-centric scenario is considered, where a FR system is embedded in a decision support tool that alerts an analyst to the presence of individuals of interest in multiple video feeds. Individuals can be added or removed from the system by the analyst, and their facial models can be refined over time with new reference sequences. In this framework, the use of concept change detection is proposed to guide an ensemble learning strategy. Each enrolled individual is modeled by a dedicated ensemble of two-class classifiers, each one specialized in a different conditions detected in reference sequences. In addition, this framework allows for a dynamic adaptation of its behavior to changing capture conditions during operations. A dynamic ensemble fusion rule is proposed, relying on concept models to estimate the relevance of each classifier w.r.t. each operational input. Finally, system decisions are accumulated over tracks following faces across consecutive frames, to provide robust spatio-temporal recognition.

In Chapter 2, concept change detection is first investigated to reduce the growth in complexity of a self-updating template-matching system for FR in video. A context-sensitive approach is proposed for self-updating, where galleries of reference images are only updated with highly-confident captures exhibiting significant changes in capture conditions. Proof of concept experiments have been conducted with a standard template matching system detecting changes

in illumination conditions, using thee publicly-available face databases. Simulation results indicate that the proposed approach allows to maintain system performance while mitigating the growth in system complexity. It exhibits the level of performance than a regular self-updating template matching system, with gallery sizes reduced by half.

In Chapter 3, a new framework for an adaptive multi-classifier system is proposed for FR in VS. It is comprised of an ensemble of incremental learning classifiers per enrolled individual, and relies on concept change detection to refine facial models with new reference data available over time while mitigating knowledge corruption. An hybrid strategy is proposed, where individual-specific ensembles are only augmented with new classifiers when an abrupt change is detected in reference data. When a gradual change is detected, knowledge about corresponding concepts is refined through incremental update of corresponding classifiers. For proof of concept experiments, a particular implementation is proposed, using ensembles of probabilistic Fuzzy-ARTMAP classifiers generated and updated with dynamic Particle Swarm Optimization, and the Hellinger Drift Detection Method for change detection. Experimental results with the FIA video surveillance database indicate that the proposed framework allows to maintain system performance over time, effectively mitigating the effects of knowledge corruption. It exhibits higher classification performance than a similar passive system, and reference probabilistic kNN and TCM-kNN systems.

In Chapter 4, an evolution of the framework presented in Chapter 3 is presented, that allows to adapt system behavior to changing operating conditions. A new dynamic weighting fusing rule is proposed for ensembles of classifiers, where each classifier is weighted by its competence to classify each operational input. Furthermore, to provide a lightweight competence estimation that doesn't interfere with live operations, classifier competence is estimated from the concept models used for change detection during training. An evolution of the particular implementation presented in Chapter 3 is proposed, where concept models are estimated with the Fuzzy C-Means clustering algorithm, and ensemble fusion is performed through dynamic weighted score-average. Experimental simulations with the FIA and ChokePoint video-surveillance datasets shows that the proposed dynamic fusion method provides a higher classification performance than the DS-OLA dynamic selection method, for a significantly lower computational complexity. In addition, the proposed system exhibits higher performance than reference probabilistic kNN, TCM-kNN and Adaptive Sparse Coding systems.

Keywords: Multi-classifier systems, adaptive face recognition, incremental learning, change detection, dynamic selection and decision fusion.

TABLE OF CONTENTS

Page

INTRODUCTION 1

CHAPTER 1 ADAPTATION OF FACE RECOGNITION SYSTEMS FOR VIDEO-SURVEILLANCE 11
1.1 Face Recognition in Video-Surveillance 11
1.1.1 Specialized Face Recognition System for Video-Surveillance 14
1.1.2 Challenges 16
1.2 Concept Change and Face Recognition 18
1.2.1 Nature of a Concept Change 18
1.2.2 Measuring Changes 20
1.3 Adaptive Biometrics 22
1.3.1 Semi-Supervised Learning 22
1.3.2 Challenges 23
1.4 Incremental Learning of Classifiers 23
1.4.1 Fuzzy-ARTMAP Networks 25
1.4.2 Challenges 28
1.5 Adaptive Ensemble of Classifiers 29
1.5.1 Generation and Update of Classifier Pools 29
1.5.1.1 Diversity Generation 29
1.5.1.2 Adaptation to Concept Change 32
1.5.2 Classifier Selection and the Fusion Rule 33
1.5.3 Challenges 35

CHAPTER 2 CONTEXT-SENSITIVE SELF-UPDATING FOR ADAPTIVE FACE RECOGNITION 37
2.1 Introduction 38
2.2 Self-Updating for Face Recognition 41
2.2.1 A General System 41
2.2.2 Adaptive Biometrics 42
2.2.3 Self-Updating Methods 44
2.2.3.1 General Presentation 44
2.2.3.2 Challenges 45
2.3 Self-Updating Driven by Capture Conditions 47
2.3.1 Framework For Context-Sensitive Self-Update 48
2.3.2 A Specific Implementation 49
2.3.2.1 A Template Matching System 49
2.3.2.2 Detecting Changes in Capture Conditions 50
2.4 Simulation Methodology 51
2.4.1 Face Recognition Databases 51

2.4.1.1 Multi-Modal Dipartimento di Ingegneria Elettrica ed Elettronica 51
2.4.1.2 CMU Faces in Action 53
2.4.1.3 Face Recognition Grand Challenge 54
2.4.2 Protocol 55
2.4.2.1 Simulation Scenario 56
2.4.2.2 Performance Measures 58
2.5 Simulation Results 58
2.5.1 Continuous User Authentication with DIEE Data 58
2.5.2 Video Surveillance with FIA Data 61
2.5.3 Unconstrained Face Recognition with FRGC Data 64
2.5.4 Summary and Discussions 66
2.6 Conclusion 67

CHAPTER 3 ADAPTIVE ENSEMBLES FOR FACE RECOGNITION IN CHANGING VIDEO SURVEILLANCE ENVIRONMENTS 71
3.1 Introduction 72
3.2 Background – Face Recognition in Video Surveillance 77
3.2.1 A generic system for video face recognition 78
3.2.2 State-of-the-art in video surveillance 79
3.2.3 Challenges 81
3.3 Concept Change and Face Recognition 82
3.3.1 Detecting changes 84
3.3.1.1 Density estimation measures 85
3.3.1.2 Thresholding 87
3.3.2 Adaptive classification for changing concepts 88
3.3.3 Synthetic test case: influence of changing concepts 91
3.4 An Adaptive Multi-Classifier System with Change Detection 94
3.4.1 Classification architecture 99
3.4.2 Change detection 101
3.4.3 Design and adaptation of facial models 104
3.4.4 Short and long term memories 107
3.4.5 Spatio-temporal recognition – accumulation of responses 107
3.5 Experimental Methodology 108
3.5.1 Video-surveillance data 109
3.5.1.1 Pre-processing 109
3.5.1.2 Simulation scenario 111
3.5.2 Reference systems 112
3.5.3 Experimental protocol 114
3.5.4 Performance measures 117
3.5.5 Memory complexity measures 119
3.6 Results and Discussion 120
3.6.1 Change detection performance 120

3.6.2 Transaction-level performance 122
3.6.3 Performance of the full system over time 127
3.7 Conclusion 129

CHAPTER 4 DYNAMIC MULTI-CONCEPT ENSEMBLES FOR VIDEO-TO-VIDEO FACE RECOGNITION 131
4.1 Introduction 132
4.2 Face Recognition in Video Surveillance 137
4.3 Adaptive Ensemble Strategies for Video to Video Face Recognition 138
4.3.1 Generation and Update of Base Classifiers 139
4.3.2 Classifier Selection and Update of Fusion Rule 141
4.4 Dynamic Multi-Concept Ensembles of Classifiers 143
4.4.1 General Framework 144
4.4.1.1 Design and Update Architecture 144
4.4.1.2 Operational Architecture 146
4.4.2 Concept Densities and Dynamic Weighting 147
4.4.3 Specific Implementation 150
4.4.3.1 Concept Densities 150
4.4.3.2 Change Detection 152
4.4.3.3 Dynamic Competence Evaluation and Decision Fusion 153
4.5 Experimental Protocol 154
4.5.1 The Faces in Action Dataset 154
4.5.1.1 Dataset presentation 154
4.5.1.2 Simulation scenario 154
4.5.2 The ChokePoint Dataset 156
4.5.2.1 Dataset presentation 156
4.5.2.2 Simulation scenario 157
4.5.3 Reference systems 157
4.5.4 Specific DMCE Implementation Parameters 159
4.5.4.1 Feature Extraction and Selection 159
4.5.4.2 Classifier Training and Ensemble Prediction 159
4.5.4.3 Concept Densities 160
4.5.4.4 Tracking and Accumulation of Predictions 160
4.5.5 Protocol for Validation 160
4.5.6 Performance Evaluation 162
4.5.6.1 Transaction-level performance 162
4.5.6.2 Trajectory-level performance 163
4.6 Results and Discussion 163
4.6.1 Transaction-level Performance 163
4.6.1.1 FIA Dataset 165
4.6.1.2 ChokePoint Dataset 166
4.6.2 Trajectory-level Performance 167
4.6.2.1 FIA Dataset 167

4.6.2.2 ChokePoint Dataset 168
4.7 Conclusion 168

CONCLUSION AND RECOMMENDATIONS 171

APPENDIX I SUPPLEMENTARY RESULTS FOR DMCE IMPLEMENTATION (CHAPTER 4) 175

BIBLIOGRAPHY 181

LIST OF TABLES

Page

Table 1.1 Types of change occurring in face recognition for video surveillance environments. 19

Table 2.1 Summary of the three experimental scenarios. 55

Table 3.1 Types of changes occurring in FRiVS environments. 83

Table 3.2 Performance for the rotating checkerboard data of a PFAM-based system updated through incremental learning and trhough the learn and combine strategy. 93

Table 3.3 Average number of ROI captured per person over 3 indoor sessions (s = 1,2,3) of the FIA database. 111

Table 3.4 Correspondence between the 9 reference video sequences used to adapt proposed AMCSs and the original *FIA* video sequences. 111

Table 3.5 Changes detected per individual of interest (marked as a X) for each update sequence. 120

Table 4.1 Correspondence between the 9 reference trajectories of the experimental scenario and the original *FIA* video sequences. 155

Table 4.2 Chokepoint verification protocol, presented in (Wong *et al.*, 2011). 157

LIST OF FIGURES

Page

Figure 1.1 General *video-to-video* face recognition system. 11

Figure 1.2 Illustration of abrupt, gradual and recurring changes occurring to a single concept over time, as defined in (Kuncheva, 2008). 19

Figure 1.3 Architecture of a FAM network. 26

Figure 2.1 General FR system trained for N individuals. 41

Figure 2.2 A FR system based on template matching that allows for self-update. 44

Figure 2.3 A template matching system that integrates context-sensitive self-updating. 48

Figure 2.4 DIEE dataset. An example of randomly chosen facial captures for two individuals. 52

Figure 2.5 FIA dataset. An example of randomly chosen facial captures for two individuals. 53

Figure 2.6 FRGC dataset. An example of randomly chosen facial captures for two individuals 54

Figure 2.7 Simulation results with DIEE dataset where the updating threshold is selected for fpr $= 0\%$. 59

Figure 2.8 Simulation results with DIEE dataset where the updating threshold is selected for fpr $= 1\%$. 60

Figure 2.9 Simulation results with FIA dataset where the updating threshold is selected for fpr $= 0\%$. 61

Figure 2.10 Simulation results with FIA dataset where the updating threshold is selected for fpr $= 1\%$. 63

Figure 2.11 Simulation results with FRGC dataset where the updating threshold is selected for fpr $= 0\%$. 64

Figure 2.12 Simulation results with FRGC dataset where the updating threshold is selected for fpr $= 1\%$. 65

Figure 3.1 A human centric system face video-to-video face recognition. 77

Figure 3.2 Illustration of abrupt, gradual and recurring changes occurring to a single concept over time, as defined in (Kuncheva, 2008). 84

Figure 3.3 Reference and operational video sequences for the synthetic test case (Kuncheva, 2004b). 92

Figure 3.4 Architecture of the proposed $AMCS_{CD}$ for FRiVS. 95

Figure 3.5 Architecture of the CD module *i*. 101

Figure 3.6 Examples of ROIs captured by the segmentation algorithm from the cameras array of 6 during the different sessions. 109

Figure 3.7 Example of accumulated predictions for individual 21 enrolled to the $AMCS_{CD}$. 118

Figure 3.8 Histograms representation of the 5th, 6th and 7th sequence of patterns for the individual 21. 121

Figure 3.9 Average overall transaction-level performance of proposed and reference systems. 123

Figure 3.10 Average transaction-level performance after learning the 9 update sequences. 125

Figure 3.11 Average memory complexity. Amount of $F2$ prototypes for the *AMCS* systems, and amount of reference patterns for VSkNN and TCM-kNN. 126

Figure 3.12 Average accumulation AUC performance after learning the 9 update sequences. 128

Figure 3.13 Accumulation AUC performance after learning the 9 update sequences. 129

Figure 4.1 General video-to-video face recognition system. 137

Figure 4.2 Architecture of a system for video-to-video FR based on the proposed DMCE framework. 144

Figure 4.3 2D projection of 2 of the 5 concept densities (1 and 4) generated by the change detection module of DMCE for individual 201 of the FIA dataset Goh *et al.* (2005). 149

Figure 4.4 Average transaction-level classification performance for the 10 individuals of interest. 164

Figure 4.5 Average trajectory-level classification performance for the 10 individuals of interest. 167

LIST OF ALGORITHMS

Page

Algorithm 2.1 Self-update algorithm for adapting template galleries. 45

Algorithm 2.2 Protocol for simulations. 57

Algorithm 3.1 Strategy to design and update the facial model of individual *i*. 97

Algorithm 3.2 Operational strategy for one individual *i*. 98

Algorithm 3.3 Specific implementation of HDDM based CD for individual *i*. 103

Algorithm 3.4 Experimental protocol for performance evaluation. 115

Algorithm 4.1 Design and update procedure for individual *i*. 145

Algorithm 4.2 Operational procedure for individual *i* over a trajectory T 147

Algorithm 4.3 Merging of concept densities $\mathscr{A}^i[t]$ and $\Omega^i_{o^*}$ 151

Algorithm 4.4 Competence computation for DMCE DS-LA OLA variants. 159

LIST OF ABREVIATIONS

AMCS	Adaptive Multi-Classifier System
$AMCS_{CD}$	Adaptive Multi Classifier System with Change Detection
$AMCS_{incr}$	Adaptive Multi Classifier System incremental
$AMCS_{LC}$	Adaptive Multi Classifier System with Learn and Combine
ART	Adaptive Resonance Theory
AUC	Area Under the ROC Curve
AUPROC	Area Under the P-ROC Curve
CCTV	Closed Circuit Television
CD	Change Detection
CM	Cohort Model
CNN	Condensed Nearest Neighbor
DIEE	Dipartimento di Ingegneria Elettrica ed Elettronica (Department of Electrical and Electronic Engineering) dataset
DMCE	Dynamic Multi-Concept Ensemble
DNPSO	Dynamic Niching Particle Swarm Optimization
DPSO	Dynamic Particle Swam Optimization
EoC	Ensemble of Classifiers
FAM	Fuzzy-ARTMAP classifier
FIA	Faces in Action database
FP	False Positives

fpr	false positive rate
FR	Face Recognition
FRGC	Face Recognition Grand Challenge
FRiVS	Face Recognition in Video Surveillance
GMM	Gaussian Mixture Model
GLQ	Global Luminance Quality index
GQ	Global Quality index
HDDM	Hellinger Drift Detection Method
kNN	k Nearest Neighbor classifier
LBP	Local Binary Pattern
LC	Learn and Combine
LTM	Long Term Memory
MCS	Multi-Classifier System
MLP	Multi-Layer Perceptron
pAUC	partial Area Under the ROC Curve
PFAM	Probabilistic Fuzzy-ARTMAP classifier
P-ROC	Precision-Recall Operation Characteristics
PSO	Particle Swarm Optimization
ROC	Receiver Operating Characteristics
ROI	Region of Interest

TP	True Positives
tpr	trupe positive rate
SSIM	Structural Similarity Index Measure
STM	Short Term Memory
SVM	Support Vector Machine
TCM-kNN	Transductive Confidence Machine - k Nearest Neighbor
UBM	Universal Background Model
VS	Video Surveillance
VSkNN	Video Surveillance k Nearest Neighbor classifier

LISTE OF SYMBOLS AND UNITS OF MEASUREMENTS

$\mathbf{a}^i[t]$	Reference pattern for individual i provided at time t
$\mathbf{A}^i[t]$	Set of reference ROI patterns extracted from $Vs^i[t]$ or $T^i[t]$
$\mathscr{A}^i[t]$	Concept density corresponding to $T^i[t]$
α	Confidence interval of the t-statistic test
$\mathbf{b}$	Tracking feature vector extracted from a ROI during operations
$b_i[t]$	Batch of data for the individual i provided at time t
$\beta_k^i[t]$	Dynamic change detection threshold for concept k and individual i
$\mathscr{C}_k^i$	Histogram model of concept k for individual i
$\mathbf{d}$, $\mathbf{q}$	Feature vector extracted from a ROI during operations
$\mathscr{D}$	Unlabeled adaptation set
$\mathscr{D}'$	Subset of $\mathscr{D}$ selected for adaptation (highly-confident patterns)
d_{Eucl}	Euclidean distance operator
$\delta_k^i[t]$	Distance between $\mathbf{A}^i[t]$ and $\mathscr{C}_k^i$
$d^i(\mathbf{q})$	Final decision produced by EoC^i for pattern $\mathbf{q}$
EoC^i	Ensemble of classifiers for individual i
F^1, F^2, F^{ab}	Layers of a FAM classifier
$\mathscr{G}'$	Updated gallery of reference templates
$\mathscr{G}_i$	Gallery of reference templates for individual i
GLQ	Global luminance quality index

GQ	Global quality index
γ_i^c	Capture condition threshold for individual i
γ_i^d	Decision threshold for individual i
$\gamma_{o^*}^i$	Cluster fusion threshold, for concept density o^* of individual i
γ_i^u	Update threshold for individual i
Γ^i	Accumulation threshold for individual i
$\mathbf{h}$	Hyper-parameter vector of a PFAM classifier
$\mathscr{H}_k^i$	History of distance measures for concept k of individual i
i	Label of an individual of interest, $i = 1, ..., N$
IC_k^i	Incremental classifier k from EoC^i
k, o	Concept index
k^*, o^*	Index of the closest concept
K^i	Number of classifiers in EoC^i
LTM^i	Long Term Memory for individual i
$\mu_{o,l}^i$	Center of cluster l, from concept density o of individual i
$n_{o,l}^i$	Number of reference patterns associated to cluster l, from concept density o of individual i
Ω_o^i	Concept density of index o, for individual i
$p(i)$	Prior probability for individual i
$p(\mathbf{a}[t]\|i)$	Class conditional probability
$\mathscr{P}_k^i$	Pool of classifiers for concept k of individual i

Ψ	Region of competence
$\mathbf{q}$	Operational ROI pattern
$\mathbf{R}_{i,j}$	Raw input ROI j form the gallery $\mathscr{G}_i$
$\mathbf{r}_{i,j}$	Reference template j form the gallery $\mathscr{G}_i$
$S_i(\mathbf{d})$	Combined matching score for pattern $\mathbf{d}$ and individual i
$s_{i,j}(\mathbf{d})$	Matching score for pattern $\mathbf{d}$ produced by template j of individual i
$s_k^i(\mathbf{q})$	Matching score for pattern $\mathbf{q}$ produced by IC_k^i
STM^i	Short Term Memory for individual i
$T^i[t]$	Facial trajectory for individual i provided at time t
$\tau_o^i(\mathbf{q})$	Competence measure relative to concept density o and input pattern $\mathbf{q}$
$tr(\mathbf{q})$	Track ID associated to pattern $\mathbf{q}$
θ^i	Decision threshold for individual i
$Vs^i[t]$	Sequence of reference ROIs for individual i provided at time t
w_0, w_1, w_2	Inertia, cognitive and social weights of PSO
$\mathbf{W}$, $\mathbf{W}^{ab}$, $\mathbf{W}^{ac}$	Internal weights of a FAM classifier

INTRODUCTION

Biometric authentication of individuals provides several advantages in terms of security over more traditional alternatives, such as password and identification card, that can be easily stolen or forged. Its various applications can be separated intro three main categories: (1) *verification*, to confirm the identify claim of a subject by only comparing his/her features to a dedicated facial model stored in the system, (2) *identification*, to retrieve a subject's identity from a set of known individuals, and (3), *screening*, to compare individuals of a potentially large crowd to a limited watchlist of individuals of interest. Among the different types of biometrics, face recognition (FR) has received a growing attention during the past decades due to its limited interaction as opposed to systems based on other biometrics such as fingerprint or iris.

A FR system aims to detect the presence of individuals of interest in images presented during operations, using facial models generated with reference data (e.g. pictures of individuals of interest). Depending on the classification algorithm, facial models can be defined in different ways. For example, with template matching systems, facial models are usually a set of one or more reference captures, to which faces captured during operations are compared. With neural networks or statistical classifiers, facial models are represented by the classifiers' internal parameters (neural network weights or distribution parameters) that are estimated during training. Other methods rely on dimensionality reduction to minimize intra-class variability, and estimate facial models as linear or non-linear manifolds of lower dimensions that embed reference data. Finally, with methods relying on sparse representations, facial models can be represented by dictionaries learned through the decomposition of the reference images.

FR systems can be divided intro three families with respect to the nature of reference and operational data (Zhou *et al.*, 2003). In *still-to-still* FR, reference data are regions of interest (ROIs) extracted from still images of individuals of interest, and the system is provided with still pictures to perform recognition during operations. In *still-to-video* FR, models are also

generated with ROIs from still images, but the system processes frames from video streams to perform recognition. Finally, *video-to-video* FR systems process frames from video streams as both reference and operational data. Continual advances in terms of processing power, video processing algorithm and the growing availability of low cost cameras motivated the development of intelligent video surveillance (VS) systems based on *still-to-video* or *video-to-video* FR algorithms.

VS networks are usually comprised of a growing number of cameras, and transmit or archive massive quantities of data for reliable decision support. In this context, where individuals of interest (criminals, terrorists, etc.) in a watchlist have to be recognized in a dense crowd of unknown people at major events or airports, the ability to perform recognition in a discrete and non-obtrusive way can be crucial. This task is presently conducted semi-automatically by analysts responsible for monitoring several camera streams, and these systems are designed as tools to assist their decision making process. For example, in *watch-list screening* applications, facial models are designed using ROIs extracted from reference still images or mugshots of a watchlist. *Still-to-video* FR is then performed on live video feeds, to alert the analyst to the possible presence of individuals of interest. Another example is *person re-identification* for search and retrieval applications, where facial models are designed using ROIs extracted from reference videos and tagged by the analyst, to perform *video-to-video* FR in either live or archived (post-event analysis) videos.

Problem Statement

This thesis focuses on *video-to-video* FR systems, as required in *person re-identification* applications, where soft biometrics can be used to track individuals over video streams or perform recognition when faces are not visible. A general human-centric scenario is considered, where an analyst has the ability to enroll specific individuals from a video feed into a watchlist to follow and monitor their whereabouts over a set of IP cameras, and then remove their models from

the system when surveillance is no longer required. The use of VS in public security organization is currently limited to human recognition capabilities, as the application of state-of-the art FR to video surveillance often yields poor performance. As a matter of fact, VS is performed under semi-controlled (e.g., in an inspection lane, portal or checkpoint entry) and uncontrolled (e.g., in cluttered free-flow scene at an airport or casino) capture conditions, which generate multiple sources of variations in the ROIs extracted from video feeds (pose orientation, scale, expression, illumination, motion blur, occlusion, etc.). Furthermore, facial models are usually designed a priori, using a limited number of high quality reference captures, and the accuracy of FR is highly dependent on the representativeness of these captures.

Several authors proposed to compensate for a lack of knowledge in facial features by using the temporal dimension of video sequences. To perform spatio-temporal recognition, these systems usually process ROIs regrouped along trajectories, that correspond to a same high quality track of an individual across consecutive frames. These methods either consider motion information as additional features (Matta and Dugelay, 2007; Liu and Cheng, 2003), or accumulate classifiers outputs over several frames to produce a more accurate prediction, based on several observations instead of one specific condition (Stallkamp *et al.*, 2007; Barry and Granger, 2007; Gorodnichy, 2005b).

Dedicated video-surveillance systems are however not very numerous in the scientific literature. This problem usually considered as *open-set*, where it is assumed that most individuals observed during operations are not enrolled to the system (Li and Wechsler, 2005). To address this problem, Li et al. proposed a TCM-kNN classifier with a dedicated reject option for the unknown individuals (Li and Wechsler, 2005).

Other contributions have been made by Ekenel, Stallkamp et al. (Ekenel *et al.*, 2009; Stallkamp *et al.*, 2007), with the use of a class-modular architecture, comprised of a specific 2-class classifier per individual (individual vs. the rest of the world). The advantages of class-modular

architectures in face recognition in video surveillance (FRiVS) (and biometrics in general) include the ease with which facial models (or individuals) may be added, updated and removed from the systems, and the possibility of specializing feature subsets and decision thresholds to each specific individual. This separation of a N-class recognition problem into N 2-class problems has already been proven beneficial in other complex applications, such as handwriting recognition (Oh and Suen, 2002; Kapp *et al.*, 2007; Tax and Duin, 2008). Moreover, it has been argued that biometric recognition is in essence a multi-classifier problem, and that biometric systems should co-jointly solve several classification tasks in order to achieve state-of-the-art performance (Bengio and Mariethoz, 2007), for example using a common universal and cohort model.

While a limited amount of reference captures is usually available for the initial design of facial models, new reference videos may become available over time, either during operations or through some re-enrollment process. These new reference captures can be used to refine facial models, possibility adding new information relevant to previously-unknown capture conditions. Different types of approaches have been proposed to address the update of biometric models over time, which can either involve *supervised* or *semi-supervised* learning, depending on the labeling process. While *supervised* learning involves a manual labeling of the reference captures (for example by the analyst), *semi-supervised* approaches rely on automatic labeling by the system. For example, De la Torre et al. (De-la Torre *et al.*, 2015) proposed a multi-classifier system learning from facial trajectories in VS, in which an ensemble of classifiers is dedicated to each enrolled individual. During operations, system predictions are accumulated along trajectories defined by a face tracker, leading to a positive recognition when their accumulation surpass a detection threshold. To update facial models, highly-confident trajectories are selected with a higher updating threshold, and then assimilated in a learn-and-combine fashion.

To adapt an individual's facial model in response to these new ROIs, the parameters of an individual-specific classifier can be re-estimated through incremental learning. For example, ARTMAP neural networks (Carpenter *et al.*, 1992) and extended Support Vector Machines (Ruping, 2001) have been designed or modified to perform incremental learning by adapting their internal parameters to new reference data. These classifiers are typically designed under the assumption that data is sampled from a static environment, where class distributions remain unchanged over time (Granger *et al.*, 2008). However, newly available data can exhibit possible changes in the underlying data distribution, for example videos captured under different observation conditions. More precisely, ROIs extracted from these video may incorporate various patterns of change that reflect varying *concepts*. In pattern recognition, a *concept* can be defined as the underlying class distribution of data captured under specific condition, in a VS context due to different pose angle, illumination, scale, etc. (Narasimhamurthy and Kuncheva, 2007). While gradual patterns of change in operational conditions are often observed (due to, e.g., ageing over sessions), abrupt and recurring patterns (caused by, e.g., new pose angle versus camera) also occur in VS. A key issue in these environments is the adaptation of facial models to assimilate captures from new concepts without corrupting previously-learned knowledge, which raises the *plasticity-stability* dilemma (Grossberg, 1988). Although updating a single classifier may translate to low system complexity, incremental learning of ROIs extracted from videos that reflect significantly different concepts can corrupt the previously acquired knowledge (Connolly *et al.*, 2012; Polikar *et al.*, 2001).

In addition to monolithic incremental classifiers, adaptive methods involving ensembles of classifiers (EoC) have also been proposed for incremental learning. They allow to exploit multiple and diverse points of view of a FR problem, and have been successfully applied in cases where concepts change in time. As a matter of fact, EoCs are well suited for adaptation in changing environments since they can manage the *plasticity-stability* dilemma at the classifier level. When new reference data are significantly different than previous ones, previously ac-

quired knowledge can be preserved by initiating and training a new classifier on the new data (Kuncheva, 2004a). For example, with methods such as Learn++ (Polikar *et al.*, 2001) and other Boosting variants (Oza, 2001), a classifier is trained independently using new samples, and weighted such that accuracy is maximized. Other approaches discard classifiers when they become inaccurate or concept change is detected, while maintaining a pool with these classifiers allows to handle recurrent change (Minku and Yao, 2012).

However, while adaptive EoCs can mitigate the effects of knowledge corruption, this gain is obtained at the expense of a growing system complexity. In addition, a diversified pool of classifiers may require an additional level of adaptation during operations for a FR system in VS. More precisely, although temporally related, a same video sequence may not contain face captures representative of the same concept (e.g. due to head movement during capture), which means that only a subset of the classifiers from a EoC would be competent for each capture condition. In some cases, classifiers specialized in significantly different capture conditions may even degrade system performance by adding incorrect predictions in the decision process. While numerous adaptive ensemble methods propose to update classifier subsets or weights depending on the observed concepts (Ortíz Díaz *et al.*, 2015; Ramamurthy and Bhatnagar, 2007), EoCs are usually updated depending their performance over a recent window of samples, only considering possible drifts toward a single concept in the input stream.

Several methods have been proposed that rely on a dynamic region evaluation using validation data, and different ways to evaluate classifier competence (Britto *et al.*, 2014). While this can improve system accuracy by preventing unrelated classifier to affect the final decision, using these methods in an adaptive ensemble that integrates newly acquired reference data, and in a changing environment, would significantly increase system computational and memory complexity. They usually require a re-evaluation of the classifiers' competence for every new

batch of reference data, as well as a costly neighbour evaluation for each input facial capture, within in a growing set of validation data stored in memory.

Objectives and Contributions

In this thesis, a new framework for *video-to-video* FR is proposed, designed to update facial models of individuals of interest using reference videos over time in a VS environment. To maintain representative and up-to-date facial models, *concept change* detection is used to guide the updating process of individual-specific ensembles of incremental classifiers. In addition, to adapt the system's operational architecture to the variability of VS sequences and only rely on relevant information during operations, a dynamic adaptation of ensembles' fusion rules is performed for each facial capture, using lightweight concept representations.

The main contributions of this thesis rely on the usage of *concept change* detection in:

a. A new *self-updating* technique, that exploits image quality measures to mitigate the growth in system complexity due to the addition of redundant information.

b. A new adaptive multi-classifier framework, that exploits change detection to guide an hybrid incremental learning strategy, relying on both incremental learning and ensemble techniques.

c. A new dynamic multi-concept ensemble framework, that models multi-modal concept densities from reference trajectories to detect abrupt changes, and dynamically adapt ensemble fusion rules to the observation conditions of each input capture.

Organization of the Thesis

This manuscript-based thesis is organized into four chapters. In Chapter 1, an overview of the literature is presented, starting with face recognition in video surveillance, and followed by

concept change detection and adaptive biometrics in general, to end with methods for incremental learning of classifiers and adaptive ensembles.

In Chapter 2, *concept change* detection is first considered to mitigate the growth in complexity of a *self-updating* template-matching FR system over time. A *context-sensitive* self-updating technique is proposed, that combines a standard *self-updating* procedure with a *concept change* detection module. In this system, the addition of a new capture into the galleries depends on two conditions: 1) its matching score is above the self-updating threshold (highly confident capture), and 2), the capture contains new information w.r.t. captures already present in the gallery (i.e. different concept). With this technique, one can avoid frequent uses of costly template management schemes, while still enhancing intra-class variation in facial models with relevant captures exhibiting different concepts. A particular implementation of this *context-sensitive* self-updating technique is presented for a basic template matching system, where changes are detected in illumination conditions. Proof-of concept experimental simulations using thee publicly-available face databases (DIEE (Rattani *et al.*, 2013), FIA (Goh *et al.*, 2005) and FRGC (Phillips *et al.*, 2005)) show that this technique enables to maintain the same level of performance than a regular self-updating template matching system, while reducing the size of template galleries by half, effectively mitigating the computational complexity of the recognition process over time. The contents of this Chapter have been submitted as a chapter to the book "Adaptive Biometric Systems: Recent Advances and Issues", by Springer.

In Chapter 3, a *concept change* detection module is embedded in a new framework for an adaptive multi-classifier system (AMCS) for video-to-video FRiVS. In this framework, in addition to reduce the growth in complexity when assimilating new reference data from previously-observed concept, *concept change* detection allows to assimilate different concepts while mitigating knowledge corruption. It is comprised of an ensemble of incremental 2-class classifier for each enrolled individual that allows to adapt their facial models in response to new ref-

erence videos, through an hybrid strategy involving either incremental learning or ensemble generation. When a new video sequence is available for update, a change detection mechanism is used to compromise between plasticity and stability. If the new data incorporates an abrupt pattern of change w.r.t. previously-learned knowledge (representative of a new concept), a new classifier is trained on the data and combined to an ensemble. Otherwise, previously-trained classifiers are incrementally updated, to refine the system's knowledge on previously-observed concepts. During operations, faces of each different individual are tracked and grouped over time, allowing to accumulate positive predictions for robust spatio-temporal recognition. A particular implementation of this framework has been proposed for validation, involving ensembles of 2-class PFAM classifiers for each individual, where each ensemble is generated and evolved using an incremental training strategy based on a dynamic PSO, and the hellinger drift detection method to detect concept changes. Experimental simulations with the FIA dataset (Goh *et al.*, 2005) indicate that the proposed framework exhibits higher classification performance than a probabilistic kNN based system adapted to video-to-video FR, as well as a reference open-set TCM-kNN system, with a significantly lower complexity. In addition, when compared to a passive AMCS where the change detection process is bypassed, the proposed active methodology allows to increase the overall performance and mitigate the effects of knowledge corruption when presented with reference data exhibiting abrupt changes, yet controlling the system's complexity as the addition of new classifiers only triggered when a significantly abrupt change is detected. The contents of this Chapter have been published in the 286th volume of "Information Sciences" (Pagano *et al.*, 2014).

In Chapter 4, an evolution of the framework presented in Chapter 3 is proposed, called Dynamic Multi-Concept Ensembles (DMCE). In addition to relying on *concept change* to guide the updating process of individual-specific EoCs with newly-available reference data, the proposed system also allows for a dynamic adaptation of EoCs' fusion rules during operations. A new dynamic weighting fusing rule is proposed, estimating, for each input, the competence of each

classifier using density models of associated concepts. To account for intra-class variations in reference trajectories and overlapping competence regions of classifiers, concepts are modeled as sets of clusters in the feature space. A particular implementation of this system has been proposed for validation, with ensemble of 2-class Probabilistic Fuzzy-ARTMAP classifiers generated and updated though a dynamic PSO strategy, and concept density representation based on Fuzzy C-means centers. Experimental simulations with two video-surveillance datasets (FIA (Goh *et al.*, 2005) and ChokePoint (Wong *et al.*, 2011)) indicate that DMCE provided a higher classification performance than dynamic selection methods, for a significantly lower computational complexity. In addition, DMCE exhibits a higher performance than a probabilistic kNN based system adapted to video-to-video FR, a reference open-set TCM-kNN system as well as an Adaptive Sparse Representation face recognition system. The contents of this Chapter have been submitted to "Transactions on Neural Networks and Learning Systems", by IEEE.

CHAPTER 1

ADAPTATION OF FACE RECOGNITION SYSTEMS FOR VIDEO-SURVEILLANCE

This thesis considers a *video-to-video* FR system embedded in a human-centric decision support tool for intelligent VS. In surveillance applications such as real-time *monitoring* or *person re-identification*, it aims to detect the presence of individuals of interest enrolled to the system by an analyst, through the analysis of one or multiple video feeds. Facial models used for detection are designed with initial reference video sequences, and may be refined over time using new sequences, either manually selected by the analyst (*supervised learning*) or automatically by the system *(semi-supervised learning*). During operations, faces detected in video feeds are matched against the facial models of the individuals enrolled to the system. For each close resemblance determined by the system, the analyst is alerted and asked for confirmation.

1.1 Face Recognition in Video-Surveillance

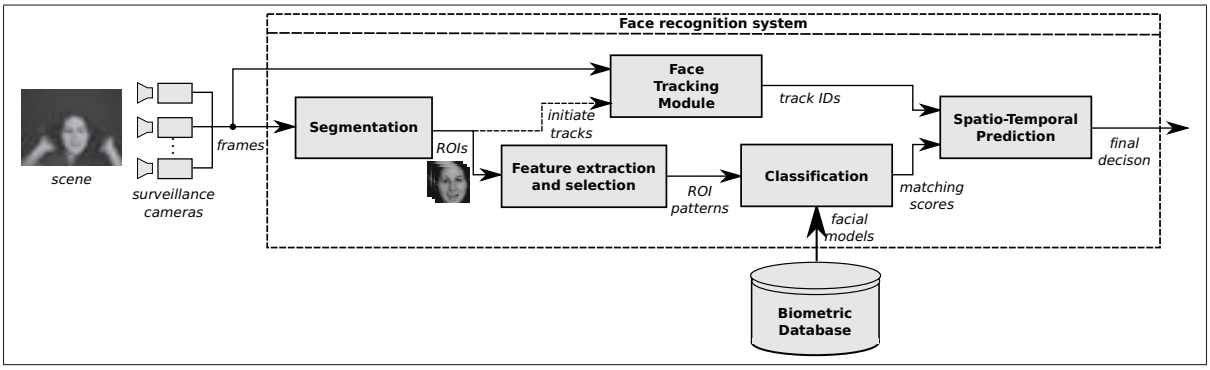


Figure 1.1 General *video-to-video* face recognition system.

Figure 1.1 presents the operational architecture of a *video-to-video* FR system. Each camera captures streams of 2D images or frames containing faces to be identified, which are first processed by a segmentation module to isolate ROIs corresponding to the actual facial regions. Then, discriminant features are extracted to generate *ROI patterns* (e.g. feature vectors). These

are are provided to the classification module, in which they are compared to facial models of individuals of interest stored into a biometric database. In parallel, tracking features are extracted to follow the position of ROIs over several frames. Tracking and classification information are then combined to generate final decisions.

The main modules are detailed in the following list:

- *Segmentation*: The segmentation task is to isolate ROIs of the raw pictures collected by the cameras, which are, in this case, the position and pixels of faces to be identified. In general, appearance-base methods have yielded higher performance in face detection applications (Yang *et al.*, 2002). These methods rely on learning strategies to build a general facial model from reference images, which can then be used to detect whether an input frame contains a face. For example, the popular Viola-Jones algorithm (Viola and Jones, 2004) relies on AdaBoost to generate classifiers selecting discriminating *haar* features in facial captures, which are combined through cascading during detection.

- *Feature extraction*: Once the facial regions are detected, discriminating features may be extracted to build feature vectors (*ROI patterns*). These are specific characteristics that help classifiers identify and differentiate the individuals to be detected, by mitigating possible sources of variability between captures from the same individual (*intra-class variability*), while enhancing variability between images from different individuals (*inter-class variability*). Various methods have been proposed to extract features from 2-dimensional (2D) image pixels. For example, the local binary patterns method (Ahonen *et al.*, 2006) mitigates illumination variations by labeling each pixel of an image with its difference between each neighborhood pixel, combined into a binary number only representing relative differences in intensity. 3D methods have also been considered to retain information relative to face geometry. These methods are either based on a depth representation with gray scale values in 2D images (called 2.5D image), or shape models such as polygonal meshes consisting of lists of points and edges. For example, active appearance models (Cootes *et al.*, 2001) combine both shape and texture information, representing faces with key points statisti-

cal distributions and grey-level appearance extracted from wrapped images. Morphable models can also be used to augment the training set with synthetic captures simulating variations in pose, generated by fitting a single 2D frontal capture on a 3D morphable face model (Blanz and Vetter, 2003). Usually, feature extraction is followed by a feature selection task, to choose subset excluding redundant and non-discriminative features that may generate noise and thus decrease the performance of the system. For instance, eigenfaces (Turk and Pentland, 1991) uses principal component analysis to reduce the dimentionality of the feature space by combining the original axes into new ones, called eigenfaces. Images can then be expressed as a linear combination of a reduced number of these new axes, only selecting the ones exhibiting highest variability.

- *Classification*: Facial models stored in the biometric database are usually designed a priori, using one or several reference *ROI patterns*, and their nature depend on the classification algorithm. For example, with a *template matcher*, facial models may be galleries of one or several reference *ROI patterns*, in which case matching scores for each operational *ROI pattern* would be computed from distance measures to these galleries. Classification may also be performed using *neural networks* (e.g multi-layer perceptrons (Riedmiller, 1994) or ARTMAP neural networks (Carpenter *et al.*, 1991)) or *statistical classifiers* (e.g. naïve Bayes classification (Duda and Hart, 1973)), in which case facial models would consist of parameters estimated during training using reference *ROI patterns* (neural networks weights, statistical distribution parameters, etc.). In addition, specific classification methods have been proposed for 3D face models. For example, matching scores can be computed from distances between sets of facial feature points fitted on a 3D morphable face model (Ansari *et al.*, 2003). Finally, depending on the nature of the classifier, its output can either be a binary decision (e.g. "*yes, the observed individual closely resembles the facial model*") or a matching score (e.g. percentage of resemblance) for each enrolled individual. In the latter case, binary decisions can be produced with application-dependent heuristics. For example, an identification system for surveillance may predict the identity of the observed individual with a maximum rule, selecting the enrolled individual with the highest

matching score, while a verification system for access control usually confirms the claimed identity by comparing the corresponding matching score to a decision threshold.

- *Tracking and spatio-temporal prediction*: *Video-to-video* FR systems usually produce their final decision based on video streams, from which a trajectory of ROIs can be extracted. More precisely, using information extracted during segmentation such as faces' positions in the scene and basic appearance characteristics, the presence of individuals can be tracked over several frames. This allows for a more robust prediction, basing the decision on a whole sequence instead of a single capture (Matta and Dugelay, 2007). With *track-and-recognition* systems, recognition (classifier outputs) and kinematic information are combined. For example, spacial distribution in feature space and temporal dynamics can be learned by a Hidden Markov Model classifier (Liu and Cheng, 2003), or combined within the same multi-modal distribution using Gaussian Mixture Models (Matta and Dugelay, 2007). On the other hand *tracking-then-recognition* methods have been proposed to track ROIs over consecutive frames, and combine individual predictions for each track to provide the final result. These methods allow to improve classification by reducing the impact of outlier captures. For example, in the *what-and-where* fusion neural network (Barry and Granger, 2007), faces are tracked using Kalman-filter banks to gradually accumulate Fuzzy-ARTMAP responses along individual trajectories.

1.1.1 Specialized Face Recognition System for Video-Surveillance

FRiVS remains a challenging task, since faces captured in video frames are typically of a lower quality than still images. Furthermore, their appearance may vary considerably due to limited control over capture conditions (e.g., illumination, pose, expression, resolution, occlusion, etc.), and changes in individuals' physiology (e.g., facial hair, aging, etc.) (Matta and Dugelay, 2009). Given these difficulties, more powerful front end processing (face capture and representation) and back-end processing (fusion or responses from cameras, templates, frames) are required for robust performance. While numerous 3D FR methods have been proposed to address such variations in facial appearance, they remain computationally intensive, and their

performance is usually dependent on the availability of high resolution ROIs, which can't be guaranteed in VS environments (Barr *et al.*, 2012).

In addition, FRiVS is usually referred to as an *open-set* problem, where it is assumed that most faces captures during operations do not correspond to an individual of interest enrolled to the system (Li and Wechsler, 2005). To address this specificity, the FRVT2002 performance test (Phillips *et al.*, 2003) proposed to add a reject option to a classification system, comparing matching scores to a detection threshold, in order to differentiate individuals of interest from unknown ones (which matching scores are usually lower).

For example, Li et al. proposed (Li and Wechsler, 2005) a face recognition system based on a modification of the kNN algorithm, the TCM-kNN classifier (transduction confidence machine kNN) developed by Proedrou et al. (Proedrou *et al.*, 2002). For each enrolled individual, a specific threshold is computed from the peak-to-side ratio of the matching scores, to identify patterns from unknown individuals from the analysis of the whole response of the classifier. Further per-individual specializations have been proposed with multi-verification systems (Stallkamp *et al.*, 2007; Ekenel *et al.*, 2009; Tax and Duin, 2008), comprised of dedicated classifiers and detection thresholds for each individual enrolled to the system. Also called class-modular architectures, these systems allow to enroll or remove individuals without requiring a complete re-initialization (in particular, re-initiating the training process for other individuals), as only one or several independent modules would be added or removed. This facilitates quick on-the-fly monitoring of the whereabouts of particular individuals in the human-centric scenario considered in this thesis. In addition, the separation of N-class into N simpler 2-class recognition problems may also improve the overall performance of the system, adopting the "divide and conquer" approach, as observed in character recognition applications using multiple multi-layer perceptrons (Oh and Suen, 2002; Kapp *et al.*, 2007).

Additional biometric applications such as speech recognition can also be related to VS, as they may also be applied in *open-set* environments. To improve system performance, *open-set* speaker identification systems (Brew and Cunningham, 2009) propose to compare target-

individual models to a Universal Background Model (UBM), a negative class generated from samples of other unknown sounds, as well as a Cohort Model (CM), a negative class representing voices from other peoples. The use of UBM and CM may be critical with class-modular architecture, as it enables to share information between classifiers, which have been shown to increase performance in multi-classifier systems (Bengio and Mariethoz, 2007). In addition, discriminative classifiers (in this case, 2-class classifiers discriminating individuals of interest v.s. the rest of the world) have been shown to outperform generative classifiers (in this case, 1-class classifiers for each individual) when a limited amount of reference data is available (Drummond, 2006).

1.1.2 Challenges

Systems for FRiVS encounter several challenges in practice, mostly related to a lack of control over the observation environment (Committee *et al.*, 2010). As opposed to many *still-to-still* applications where observation conditions of the individuals are usually normalized, VS involves significant variations in facial appearance in video streams, which can be organized in two categories:

- Variations in interactions between individuals and cameras, such as camera angle, distance between individuals and camera, direction and intensity of movement. These can generate ROIs with multiple resolutions, intensities or directions of motion blur, and facial pose orientations.
- Variations in capture conditions, such as scene illumination or partial occlusion due to foreground objects. These can directly affect facial appearance in the 2D frames provided by the camera, by hiding or modifying facial features.

As facial models for FR systems are typically designed during an a priori enrollment phase using limited number of reference *ROI patterns*, they often poor representatives of faces to be recognized during operations (Rattani, 2010). For example, an individual may be observed in

operational video streams under a significantly different camera angle from those used to collect reference captures, and thus exhibit a facial appearance non-represented in its facial model. While class-modular architectures with reject option and UBM may increase the system's discriminative power, they still rely on incomplete representations of the recognition problem provided by initial reference data.

Although initially limited in number and representativeness, new reference video sequences may become available over time in a human-centric VS scenario. For example, in a *live monitoring* application, new footage of individuals of interest observed with different cameras may become available to the analyst after initial enrollment. These can be used to refine individual facial models, either through *semi-supervised* (operational sequences validated by the system) or *supervised* (sequences provided by another agency) learning, and increase their intra-class variability. While pattern recognition techniques for *incremental learning* may be applied to assimilate newly available data over time, a particular care must be taken when updating the system with captures from different conditions. For a FR system to remain reliable over time, key issue is the adaptation of facial models to assimilate information relative to newly-available observation conditions, but without corrupting the previously-acquired knowledge. In other words, updating a FR system with reference captures from a new camera angle shouldn't override its past knowledge about other angles, as it may still be relevant should individuals be observed under similar conditions in future video steams. This issue is called the *plasticity-stability* dilemma (Carpenter and Grossberg, 1987), i.e. the trade-off between the ability to adapt to new data and the preservation of previous knowledge.

Finally, a FRiVS system comprised of facial models representative of multiple capture conditions should be able to adapt its behaviour dynamically during operations. More precisely, ROIs extracted from a same video stream may exhibit significant and abrupt changes in facial appearance, for example due to rapid head movement of an observed individuals. In such conditions, only a subset of the system's knowledge would be relevant to each ROI, and the remain could even be harmful to a correct decision.

1.2 Concept Change and Face Recognition

Video sequences captured in a VS environment are subject to a wide range of variations, from minor fluctuations due to camera noise to significant changes in observation conditions. As a consequence, the facial appearance of detected individuals is likely to exhibit significant changes from one reference sequence to another. In pattern recognition, a *concept* is defined by the underlying data distribution of a problem at some point in time in the feature space (Narasimhamurthy and Kuncheva, 2007). *ROI patterns* extracted from facial images captured under similar conditions can thus be considered as representing similar *concepts*. This section presents an overview of the study of *concept change* for FRiVS applications.

1.2.1 Nature of a Concept Change

In pattern recognition, and more specifically FR, a statistical classification problem may change due to variations in prior, class-conditional or posterior probabilities, as a result of an evolution of the underlying data distribution of the different classes in the feature space (Kuncheva, 2004a). The main assumption is the uncertainty about the future, the data distribution from which the future instance is sampled is unknown. More precisely, *concept change* encompasses various types of noise, trends and substitutions in the underlying data distribution associated with a class (individual) or concept. A categorization has been proposed by Minku et al. (Minku *et al.*, 2010), based on severity, speed, predictability and number of re-occurrences, but four categories are mainly considered in the literature: noise, abrupt, gradual and recurring changes (Kuncheva, 2008).

Table 1.1 provides a summary of the different types of changes that can occur in a FRiVS environment, related to variations in observation conditions and individuals' physiology. From a perspective of any biometric system, changes may originate from phenomena that are either static or dynamic in nature. While a dynamic environment may generate changes through the evolution of observation conditions or individuals' physiology, changes can also be observed in a static environment with *hidden contexts*, where concepts already present in the

Table 1.1 Types of change occurring in face recognition for video surveillance environments.

Type of change	Examples in face recognition
Static environment with:	
– random noise	– inherent noise of system (camera, matcher, etc.)
– hidden contexts	– different known view points from a camera or of a face (e.g. illumination of images, new face pose or orientation) (Figure 1.2 (a))
Dynamic environment with:	
– gradual changes	– aging of user (Figure 1.2 (b))
– sudden abrupt changes	–new unknown view points on traits; change of camera (Figure 1.2 (a))
– recurring contexts	– unpredictable but recurring changes in capture conditions (e.g. lighting changes due to the weather) (Figure 1.2 (c))

observed scene are yet to be modeled in the system because of the limited representativeness of previously-observed reference captures. Figure 1.2 illustrates these types of change as they may be observed over time for a concept in a 2 dimensional space, assuming that it is observed at discrete time steps. It also shows the progression of a corresponding change detection measure.

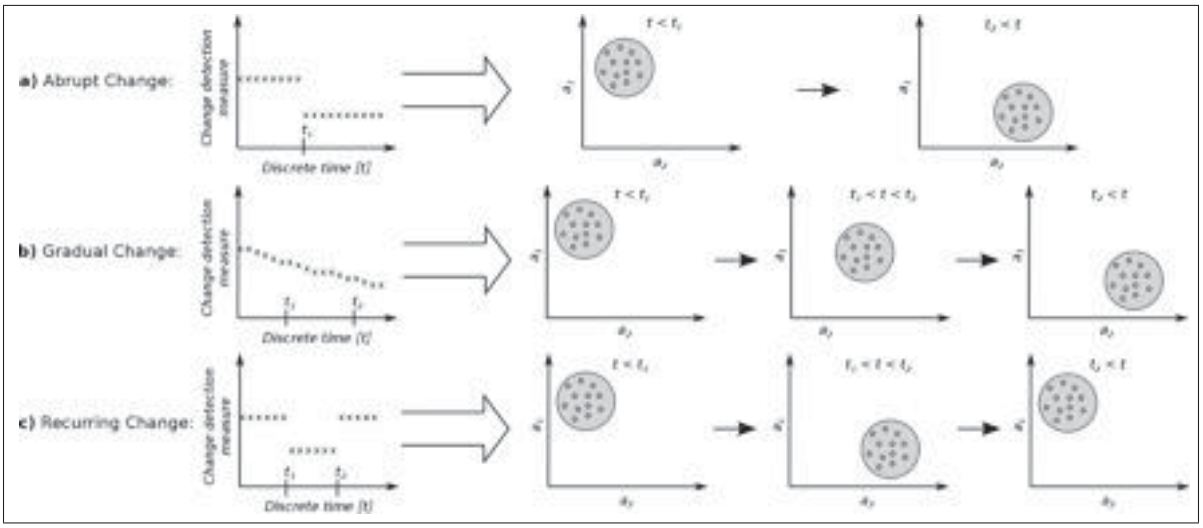


Figure 1.2 Illustration of (a) abrupt, (b) gradual and (c) recurring changes occurring to a single concept over time, as defined in (Kuncheva, 2008). The first column presents an example of the evolution of values of a change detection measure, corresponding to variations to the 2-D data distribution to the right.

In the VS scenario considered in this thesis, concept changes can be observed in reference ROI patterns extracted from newly available sequences, either caused by *hidden contexts* or the natural evolution of the observed environment. More precisely, for an individual of interest already enrolled to the system, new reference sequences with faces captured under different illumination conditions or a different pose angles correspond to abrupt changes (Fig. 1.2 (a)), and thus the addition of new concepts. On the other hand, reference sequences with faces captured under previously-encountered conditions may correspond to gradual changes (Fig. 1.2 (b)), and thus be used to refine knowledge about previously-observed concepts. Finally, a recurring change may occur when specific observation conditions may be re-encountered in the future (Fig. 1.2 (c)), which can be considered as gradual change w.r.t. a previously-encountered concept.

1.2.2 Measuring Changes

To detect possible occurrences of *concept change*, several families of measures have been proposed in the literature. They can be organized into three categories: signal processing, classification performance and density estimation in feature space.

Prior to feature extraction, *signal quality* measures have been used to accept, reject, or reacquire biometric samples, as well as to select a biometric modality, algorithm, and/or system parameters (Sellahewa *et al.*, 2010). In FRiVS, change detection can be performed by monitoring the values of an image-based quality over time. For example, several standards have been proposed to evaluate facial quality, such as ICAO 9303 (Doc, 2005), which cover image and face specific qualities. Other face quality measures compare input ROIs against facial references to assess image variations or distortions.

Change detection mechanisms using *classifier performance* indicators have also been considered for supervised learning applications (Kuncheva, 2004b). For instance, changes can be detected in system performance using accuracy, recall or precision measures on the input data (Gama *et al.*, 2004), or in the performance of a separate classifier dedicated to change detection,

trained with the data corresponding to the last known change (Alippi *et al.*, 2011). However, while directly monitoring system performance is a straightforward way to measure concept changes, it can also have several drawbacks. Relying on classifier performance for change detection may require a considerable amount of representative training data, especially when a classifier must be updated (Alippi *et al.*, 2011).

Although it may provide the most insight, detecting changes in the underlying distribution is very complex in the feature space. To reduce the computational complexity of change detection in the input feature space, several authors proposed to estimate and monitor *densities of data distribution*. These techniques rely on fitting a statistical model to the previously-observed data, which distribution in the feature space is unknown, and then applying statistical inference tests to evaluate whether the recently-observed data belong to the same model. For example, clustering methods such as *k*-means or Gaussian mixture models (GMMs) may provide a compact representation of input data distributions in the feature space (Kuncheva, 2009). Non parametric models have also been considered, such as histogram representations (Dries and Rückert, 2009; Ditzler and Polikar, 2011), which enables to avoid assumptions regarding the nature of underlying distributions. From these models, changes can then be quantified as variations in: 1) *likelihood* measures of new data w.r.t. stored models (Kuncheva, 2009), 2) *model parameters* such as cluster centers and covariance matrices of k-means models or GMMs (Kuncheva, 2009) or polynomial regression parameters (Alippi *et al.*, 2011, 2013), or 3), *density distance* measures, such as hellinger (Ditzler and Polikar, 2011) or binary distances (Dries and Rückert, 2009) between histogram models.

Density estimation methods provide a lower level information than classifier performance indicators, and and will be considered for *concept change* detection in this thesis. As a matter of fact, performance indicators of classifiers trained over previously-encountered data are merely a consequence of possible changes in the underlying data distribution, and may be subject to a bias caused by their training procedure. On the other hand, density estimation methods directly reflect the structure of underlying distributions.

1.3 Adaptive Biometrics

As presented in Section 1.1.2, the lack of representative reference captures during the initial enrollment of a FRiVS system may be addressed by the ability to update facial models over time. Such adaptation can either be *supervised* or *semi-supervised*, depending on the labeling process of the reference data (Zhu, 2005). A *supervised* learning scenario for FRiVS involves updating facial models of enrolled individuals with reference sequences of the same individuals, which identity have been manually confirmed (for example by an analyst). On the other hand, a *semi-supervised* learning system would automatically update its facial models with operational sequences labeled by its own decision mechanism.

While supervised adaptation may represent an ideal scenario with an error-free labeling process, human intervention is often costly or not feasible (Rattani *et al.*, 2009). Depending on the application, the ability to perform semi-supervised adaptation may be the only viable solution, which has lead to the development of various strategies to increase the robustness of such systems, such as *self-update* (Jiang and Ser, 2002; Ryu *et al.*, 2006) and *co-update* techniques (Rattani *et al.*, 2009, 2008).

1.3.1 Semi-Supervised Learning

As *semi-supervised* learning relies on system decision to label reference data, the stability of its performance over time strongly relies on its initial classification performance. Updating the system with mislabeled captures could have dramatic consequences, as the corruption of facial models would affect the accuracy for the corresponding individuals, leading to even more mislabeled reference captures, and so on (Rattani *et al.*, 2009, 2013).

To prevent this behaviour, *self-update* methods (Jiang and Ser, 2002; Ryu *et al.*, 2006; Rattani *et al.*, 2011) propose to update facial models with only highly-confident operational captures. For example, consider a FR system based on template matching, with galleries $\mathscr{G}_i$ of reference *ROI patterns* for each enrolled individual i as facial models. During operations, each *ROI pattern* $\mathbf{q}$ extracted from the video stream is compared to these galleries, to compute its *match-*

ing score $s_i(\mathbf{q})$ for each individual of interest. These scores are then compared to a decision threshold γ^d, to determine whether to label $\mathbf{q}$ as belonging to individual i ($s_i(\mathbf{q}) \geq \gamma^d$) or not ($s_i(\mathbf{q}) < \gamma^d$). In *self-update* systems, an additional stricter updating threshold γ^u is considered (usually $\gamma^u \geq \gamma^d$), to only select captures with high degree of confidence for updating, i.e. for which $s_i(\mathbf{q}) \geq \gamma^u$.

1.3.2 Challenges

While *self-update* methods have been shown to improve system accuracy over time, it has been argued that updating with only highly confident captures may result in the addition of redundant information in the galleries, and thus a marginal gain in performance at the expense of a considerable increase in system complexity (Rattani *et al.*, 2009). In addition, operational samples with more drastic changes are less likely to generate classification scores surpassing the updating threshold, preventing the classification system to assimilate this new information. To address this limitation, *co-updating* methods have been proposed to benefit from complementary biometric systems (Rattani *et al.*, 2009, 2008). Each system is initialized with reference patterns from a different source (or different features extracted from the same source), and performs classification of operational input data. In the same way as self-updating techniques, each system selects highly-confident samples based on an updating threshold, but this information is also shared with other systems. If the classification score of one system surpasses its updating threshold, the others will also consider the corresponding samples as highly confident, and perform adaptation. While *co-updating* is usually applied with multiple biometric traits, it could also be applied in, for example, a FRiVS scenario involving multiple cameras. In this situation, relying on multiple point of views could mitigate the effect of disruptions such as motion blur that would be less likely to affect every camera at the same time.

1.4 Incremental Learning of Classifiers

In either *supervised* or *semi-supervised* learning applications, the refinement of classifiers' facial models over time fall within the category of incremental learning. A critical property of

incremental classifiers is the ability to learn new information without corrupting previously acquired knowledge (*catastrophic forgetting*). This issue is referred to as the *stability-plasticity dilemma* (Carpenter and Grossberg, 1987), the trade-off between the ability to adapt to new data and the preservation of previous knowledge. According to Polikar (Polikar *et al.*, 2001), an incremental algorithm must fulfill four conditions: 1) allow to learn additional information from new data, 2) do not require access to the previous training data to perform the update, 3) preserve previously acquired knowledge, and 4), accommodate to new classes possibly introduced by new data.

These requirements are directly related to the VS problem considered in this thesis. First of all, facial models of individuals of interest should be refined over time with newly available reference data, but without forgetting previously-encountered concepts as they may still be relevant for future operations (e.g., learning knowledge about a new facial pose shouldn't mean that others will never be observer again in the future). In addition, controlling system computational and memory complexity is critical to maintain its ability to perform live detection, hence the necessity to avoid the accumulation of past reference data. Finally, the analyst should be able to add or remove individuals of interest without requiring a re-initialization of the system

In pattern recognition, the different approaches to perform incremental learning can be separated in the following three categories:

a. Adaptations of popular pattern classifiers, such as support vector machine (SVM) (Ruping, 2001), multi-layer perceptron (MLP) (Chakraborty and Pal, 2003) and radial basis function (RBF) (Okamato *et al.*, 2003) network, to incremental learning.

b. Classifiers designed to perform incremental learning, such as ARTMAP neural networks (Carpenter and Grossberg, 1987) and Growing Self-Organizing (Fritzke, 1996) families of neural networks.

c. Architectures based on Ensembles of Classifiers (EoCs).

A key feature of these methods is their ability to pursue the adaptation of their parameters without requiring access to previous data, as stated in (Polikar *et al.*, 2001). For example, incremental learning with SVMs (Ruping, 2001) is achieved through data compression. Previous batches of data are only represented by the support vectors of the SVMs, a lighter representation that can be combined with newly available data to update system knowledge by training new SVMs. This method also ensures that previously-acquired knowledge is preserved, by increasing the cost of miss-classification of previous support vectors during training. Similarly, growing self-organizing networks (Fritzke, 1996) rely on unsupervised clustering, to represent reference data as categories in the feature space. When new reference data become available, these categories can be adapted, or new ones created, effectively increasing classifiers intra-class variability while preserving previously-acquired knowledge.

While architectures for EoCs are detailed in Section 1.5, the remaining of this section focuses on the particular case of ARTMAP neural networks. ARTMAP reefers to a family of neural network architectures based on Adaptive Resonance Theory (ART) that is capable of fast, stable, on-line, unsupervised and supervised, incremental learning, classification and prediction (Carpenter and Grossberg, 1987). They combine an ART unsupervised neural network with a map field. In this thesis, ARTMAP networks, and more specifically the fuzzy-ARTMAP and probability fuzzy-ARTMAP variants have been considered for their unique solution to the *stability-plasticity dilemma*. ART networks adjust previously-learned categories in response to familiar inputs, and creates new ones to accommodate to inputs different enough from those previously categorized.

1.4.1 Fuzzy-ARTMAP Networks

Several ARTMAP networks have been proposed to improve the performance of these architectures. Among them, the fuzzy-ARTMAP (FAM) (Carpenter *et al.*, 1992) integrates the fuzzy ART to process both analog and binary input patterns with the original ARTMAP architecture. This popular neural network has been designed to perform supervised incremental learning, and respects the conditions from (Polikar *et al.*, 2001). In learning mode, the sequential learn-

ing process grows the number of recognition categories according to a problem's complexity. The vigilance and match tracking process provide the mechanisms to control the local impact of new data on the existing knowledge structure.

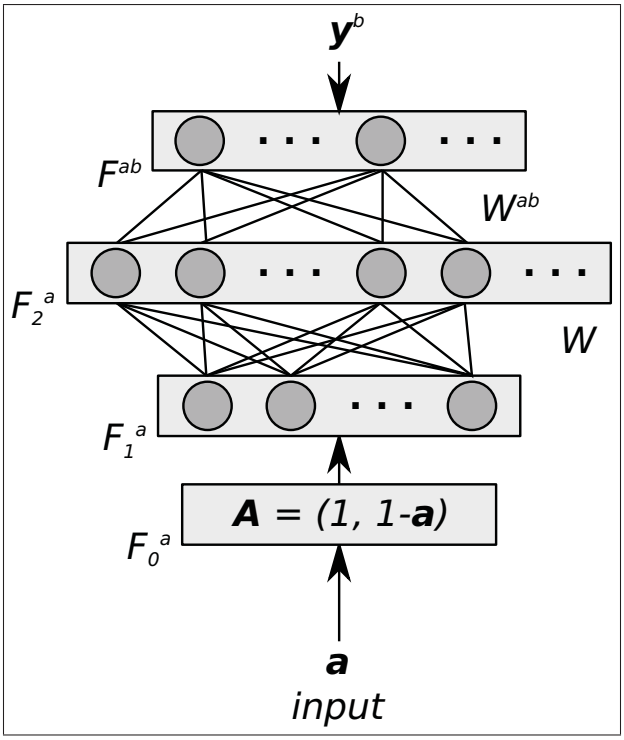


Figure 1.3 Architecture of a FAM network.

A FAM network, presented in Figure 1.3, is composed by three layers: (1) an input layer F_1, composed by $2M$ nodes (M being the dimension of the feature space), (2) an activation layer, F_2, where each node (N being the number of F_2's nodes) is associated to a category in the feature space, and (3) a mapping field, F^{ab}, of L nodes (for L classes), linking the categories of F_2 to the real world's classes. Connections between F_1 and F_2 are represented by a set of real weights $\mathbf{W} = \{w_{ij} \in [0,1] : i = 1,2,...,M; j = 1,2,...,N\}$, each category j adjusting a prototype vector $\boldsymbol{w}_j = (w_{1j}, w_{2j}, ..., w_{Mj})$. The F_2 layer is also connected to the F^{ab} layer through the binary weight set $\boldsymbol{W}^{ab} = \{w^{ab}_{jk} \in 0,1 : j = 1,2,...,N; k = 1,2,...,L\}$, the vector $\boldsymbol{w}^{ab}_j = (w^{ab}_{j1}, w^{ab}_{j2}, ..., w^{ab}_{jL})$ representing the link between the node (or category) j of F_2 and one of the L classes.

During training, a FAM network behavior relies on four hyper-parameters: (1) $\alpha > 0$, the choice parameter, (2) $\beta \in [0,1]$, the learning rate, (3) $\bar{\rho} \in [0,1]$, the baseline vigilance parameter, and (4) $\varepsilon = 0^+$, the match-tracking parameter. A categorization of the feature space is realized in the first time, the amount N of categories being determined on-line. To do so, the input vector $\mathbf{a}$ is first normalized such as $a_i \in [0,1], (i = 1...M)$, while the internal vigilance parameter ρ is initialized to the value of the baseline $\bar{\rho}$. Then, the F_1 layer compute the complementary $\mathbf{A}$ of $\mathbf{a}$ such as $\mathbf{A} \equiv (\mathbf{a}, \mathbf{a}^c)$, where $a_i^c \equiv (1 - a_i)$, which enables the computation of the activation of each node j of the F_2 layer, following the Weber-choice function: $T_j(\mathbf{A}) = |\mathbf{A} \wedge \boldsymbol{w}_j| / (\alpha + |\boldsymbol{w}_j|)$.

The node with the highest activation $J = argmax\{T_j : j = 1,...,N\}$ is then chosen, and the similarity between $\mathbf{A}$ and $\boldsymbol{w}_J$ is verified with the vigilance test $|\mathbf{A} \wedge \boldsymbol{w}_J| / 2M \geq \rho$. If the test is not satisfied, the node J is deactivated (activation put to 0), and the network tries with the second best node of F_2. When a suitable node J is found, a prediction test is realized from the vector $\boldsymbol{w}_J^{ab}$ to determine the predicted class $K = k(J)$. If the prediction is wrong, the Match-Tracking procedure deactivates the node J and changes the value of $\rho = (|\mathbf{A} \wedge \boldsymbol{w}_J| / 2M) + \varepsilon$. If the node passes the prediction test, its category is updated though the prototype vector $\boldsymbol{w}'_J = \beta(\mathbf{A} \wedge \boldsymbol{w}_J) + (1 - \beta)\boldsymbol{w}_J$. On the other hand, if no suitable node is found in F_2, a new node is created and connected to the class K by putting w_{Jk}^{ab} to 1 is $k = K$, and 0 otherwise.

Once the internal parameters (or weights) have been determined, the prediction of the class of a new input is realised by computing the activations of the F_2 layer in order to find the winning node J, and then obtain the class $K = k(J)$ linked tough the F^{ab} layer.

While FAM networks only generate binary decisions, a higher level of detail can be obtained with Probabilistic Fuzzy ARTMAP (PFAM) networks (Lim and Harrison, 1995). These are probabilistic adaptations of FAM networks that generate Gaussian distributions from the categories created though standard FAM training, thus allowing to benefit from the advantages of FAM networks while generating probabilistic outputs. More precisely, PFAM implements the following modification over the original FAM algorithm:

a. Estimation of the prior probabilities of the classes: the categories are linked with several classes, through the $\boldsymbol{W}^{ab}$ weights. They are incremented at each assignation, making possible, at the end of the training, the computation of the assignation frequency of each class k $S_k = \sum_{j=1}^{N} w_{jk}^{ab}$, as well as they prior probabilities $p(k) = S_k / \sum_{l=1}^{L} S_l$.

b. Estimation of the categories' centres: the center $\boldsymbol{w}_j^{ac}$ of each category j is the center of gravity of all the samples it contains. It's updated through $\boldsymbol{w'}_j^{ac} = (\mathbf{a} - \boldsymbol{w}_j^{ac})/|\boldsymbol{w}_j^{ab}|$ when a new sample is attributed to the category. When a new category is created, it's center is initialized at the position of the input sample $\mathbf{a}$.

c. Estimation of the categories' variances: when the training is over, the covariance matrices $\mathbf{S}$ of the categories' distributions are estimated. To limit the model complexity, they are considered as diagonal and uniform (for example, for the category j, $S_j = \sigma_j.\mathbf{I}$). For each category, only one variance value has to be computed, and is approximated with a new hyper-parameter, r, the smoothing parameter, following $\sigma_j = \min_{1 \leq i \leq N, i \neq j} ||w_j^{ac} - w_i^{ac}||/r$. It's an approximation which preserves the neighbourhood of each category.

On the other hand, Probabilistic FAM's prediction process is completely different than FAM's: for an input $\mathbf{a}$, the activation of each category j is computed as a Gaussian density, following Equation 1.1.

$$g_j(\mathbf{a}) = e^{-(\mathbf{a}-\mathbf{w}_j^{ac})^T(\mathbf{a}-\mathbf{w}_j^{ac})/2\sigma_j^2} / (2\pi)^{m/2}\sigma_j^m \tag{1.1}$$

The conditional probabilities of each class k are then determined with the "Parzen-Windows" theory, following $p(a|k) = \sum_{j \in C_k} g_j(\mathbf{a})/|w_j^{ab}|$, C_k being the set of categories associated to the class k. The posterior probability, which will be the likelihood scores, are finally estimated using the Bayes' formula: $p(k|a) = p(k).p(a|k)$.

1.4.2 Challenges

While incremental learning classifiers may allow to update facial models over time, the incremental learning of significantly different and noisy data has been shown to degrade previously-

acquired knowledge (Polikar *et al.*, 2001). For example, in ARTMAP networks, this can lead to a proliferation of category neurons on the hidden layer, causing a reduction in discrimination for older concepts and an increased computational complexity.

1.5 Adaptive Ensemble of Classifiers

EoC methods rely on the combination of several classifiers to improve overall system accuracy. Their design usually involve three steps: (1) generation of a pool of base classifiers, (2) selection of classifiers from this pool, and (3), design of the ensemble fusion rule to output a single decision (either through the selection of a single most relevant classifier or the combination of individual responses) (Kuncheva, 2004a; Britto *et al.*, 2014). EoCs methods have numerous advantages, that are of a definite interest for FRiVS applications. First of all, relying on a diversified pool of classifiers have been shown to improve overall system accuracy (Rokach, 2010), especially in complex environments with ill-defined problems. Furthermore, numerous ensemble methods proposed to perform incremental learning in changing environments have been shown to preserve previously-acquired knowledge, and yet remain adaptable to new information (Ortíz Díaz *et al.*, 2015; Polikar *et al.*, 2001; Ramamurthy and Bhatnagar, 2007). For example, instead of updating a single incremental classifier with newly-available data, training and adding new ones to an ensemble may allow to mitigate knowledge corruption, as knowledge about previously-encountered concepts would remain intact in the other classifiers of the ensembles.

1.5.1 Generation and Update of Classifier Pools

1.5.1.1 Diversity Generation

To improve system performance, a certain level of diversity is required among the combined classifiers (Brown *et al.*, 2005). In this context, a diverse EoC usually means that is its comprised of classifiers with different perceptions of the recognition problem, and multiple methods have been proposed o generate diverse ensembles. For example, classifiers can be trained

using different subsets of data, generated with k-fold data split, boostrapping or bagging techniques (Kuncheva, 2004b). Diversity can also be generated through variation of feature representations. In the random subspace approach (Ho, 1998), each classifier is trained with the same dataset, but projected into a different subspaces. Furthermore, heterogeneous ensembles can be generated by combining different types of classifiers, or similar classifiers initialized with different hyper-parameters.

In this thesis, population-based evolutionary algorithms such as Particle Swarm Optimization (PSO) and its derivatives have been considered to generate performing ensembles of classifiers. Considering a PFAM classifier which relies on 6 hyper-parameters, the optimization of these parameters through evolutionary algorithm promotes diversity inside the solution space. As the hyper-parameter (or solution) space defines the architecture of classifiers, this leads to an explicit generation of diversity in classifiers hypotheses. Empirical results from Connolly (Connolly *et al.*, 2010a) with incremental learning of PFAM classifiers for face recognition using dynamic PSOs have pointed out the correlation between diversity among the solutions in the hyper-parameter space and diversity of ensemble classifiers.

PSO is a stochastic optimization method based on the evolution of a population of solutions developed by Eberhart and Kennedy (Eberhart and Kennedy, 1995), inspired by the social behaviour of a bird's flock. Each individual (or particle) of the population (or swarm) is a possible solution (a set of parameters), and the algorithm determine their trajectory in order to maximize one or several objectives fixed by the user. In the context of the optimization of classifiers' hyper-parameters, the objective can be the classification rate of a validation database by classifiers trained with the parameters of the particles.

More precisely, let $\mathbf{p}$ be a particle in the optimization space, and thus a vector of classifier hyper-parameters. For each iteration of the optimization algorithm, until a user-defined stopping criterion (for example a maximum amount of iteraions I_{max}), the trajectory of $\mathbf{p}$ for the next iteration is governed by the speed equation:

$$v = W.v + r_1.c_1.(\mathbf{p} - \mathbf{p}_{best}) + r_2.c_2.(\mathbf{p} - \mathbf{g}_{best}) \tag{1.2}$$

The general idea is for it to evolve towards the best particle (in terms of the objective function) of the swarm $\mathbf{g}_{best}$ determined at each iteration, its best known position $\mathbf{p}_{best}$ (or particle memory), as well as its previous speed to maintain inertia. The parameters W, c_1 and c_2 are fixed by the user, and can influence the general behaviour of the swarm, while the random parameters r_1 and r_2 prevent premature convergence toward a possibly sub-optimal position trough a random weighing which ensures that all the particles are not headed toward the same general direction.

To generate ensembles of diverse classifiers, sub-swarm variants of PSO are of a particular interest, as they rely on the formation, explicit or not, of subsets of particles evolving separately (Connolly *et al.*, 2010b). This modification leads to two main consequences regarding the evolution of the swarm: (1) leader selection only considers particles inside the subset of each particles, which prevents particles from trying to get closer to leader located too far from their position (usually resulting in as useless flight), and (2), each subset evolve in a different region of the solution space, which explicitly encourages diversity in particle's parameters, thus preventing early convergence toward a local optima. For example, the Dynamic Niching PSO (Nickabadi *et al.*, 2008b) has been proposed to generate sub-swarms automatically, using a niching strategy. In this method, each particle is fist linked to an initial leader in its neighbourhood (best performing particle in a fixed-size raduis). This leader is then replaced by a new one, better than the old one in its neighbourhood, and this process is repeated until a particle which is it's own leader is found (no particle is better in its neighbourhood). All the particles of the swarm are then regrouped with the ones sharing the same leader, thus forming dynamic the sub-swarms. This processed is repeated for each iteration to avoid using the same sub-swarm topology for the entire process. Finally, particles without any sub-swarm (self-leaders only) are considered as free particles, participating to the generation of diversity among solution due to the lack of constraints, as they can still generate new sub-swarms in the following iterations. By explicitly promoting parameter diversity among local best (best particles of each subswarm), this method allows for the generation of diverse ensembles, comprised of optimal classifiers for different regions of the hyper-parameter space.

1.5.1.2 Adaptation to Concept Change

In this thesis, EoC methods are considered to adapt facial models with newly-available data. Ensemble adaptation strategies can be divided into three general categories (Kuncheva, 2004a):

a. *horse racing* methods, which train monolithic classifiers beforehand, and only adapt the combination rule dynamically (Blum, 1997; Zhu *et al.*, 2004).

b. methods using new data to update the parameters of EoCs classifiers in *incremental learning* (Gama *et al.*, 2004; Connolly *et al.*, 2013).

c. hybrid approaches, adding new base classifiers as well as adapting the fusion rule, such as the Learn++ algorithm family (Muhlbaier and Polikar, 2007; Muhlbaier *et al.*, 2009; Polikar *et al.*, 2001).

However, *horse racing* and *incremental learning* EoC approaches cannot accommodate to significantly different data. With the former, classifiers cannot update their problem representation to represent new concepts, and with the latter, they are subject to *knowledge corruption*. On the other hand, *hybrid* approaches provide a compromise between *stability* and *plasticity* to new data. Classifiers trained on previously acquired data remain intact, while new classifiers are trained for new reference data. For example, in the Learn++ algorithm (Polikar *et al.*, 2001), an ensemble is incrementally grown using, at each iteration, a weight distribution giving more importance to reference samples previously mis-classified, thus generating new classifiers specialized on the most difficult samples.

Following the definition of Gama et al. (Gama *et al.*, 2004) and Ditzler et al. (Ditzler and Polikar, 2011), ensemble methods can also be differentiated by the way they handle concept change. On one hand, *passive* methods are designed to continuously adapt to new data without monitoring possible changes, that are handled through automatic adaptation mechanisms. For example, when a new batch of data become available, boosting methods from the Learn++ family (Muhlbaier and Polikar, 2007; Muhlbaier *et al.*, 2009; Polikar *et al.*, 2001) propose to

generate one or several new classifiers, and combine them with previous ones through weighted majority voting. To adapt the system to current concepts, the voting weights are regularly adapted depending on the ensemble performance on the previous data batches. On the other hand, *active* methods monitor data streams to detect concept drifts, in which case specific adaptation mechanisms are activated. For example, Minku et al. (Minku and Yao, 2012) proposed the Diversity for Dealing with Drifts algorithm, which maintains two ensembles with different diversity levels, one low and one high, in order to assimilate a new concept emerging in the observed data. When a significant change is detected though the monitoring of the system's error rate, the high diversity ensemble is used to assimilate new data and converge to a low diversity ensemble, and a new high diversity one is generated and maintained through bagging. Alippi et al. (Alippi *et al.*, 2013) also proposed a Just-in-Time classification algorithm, using a density-based change detection to regroup reference samples per detected concept, and update a on-line classifier using this knowledge when the observed data drift toward a known concept. Other methods such as proposed by Ramamurthy et al. (Ramamurthy and Bhatnagar, 2007) and Dìaz et al. (Ortíz Díaz *et al.*, 2015) rely on concept change detection to decide whether to train a new classifier on recent data, or leave the ensemble unchanged. A new classifier is added only if a new concept is detected in the observed data, which limits unnecessary system growth with redundant information.

1.5.2 Classifier Selection and the Fusion Rule

The fusion of classifiers outputs is an example of *single biometric multiple classifiers* fusion, as presented by Ross (Ross and Jain, 2003). Such fusions are usually performed at the score, rank or decision levels. For example, the fusion in score level can be static considering several fixed rules, such as the median, the product, the minimum or the maximum score (Ulas *et al.*, 2009). These rules can also be updated at regular intervals. In *horse racing* ensemble algorithms (Blum, 1997; Zhu *et al.*, 2004), a static ensemble of L classifiers are associated with weights that are updated over time depending on their performance over past data. These are used to perform fusion through weighted majority, or to perform selection by using the prediction

of the classifier with the highest weight as the ensemble decision (Hedge β method). Another example is the *Winnow* algorithm (Littlestone, 1988), that only updates a classifier weight when it gives a correct prediction despite the ensemble decision being wrong (promotion step).

Other methods combine strategies to update both the pool of base classifiers and the fusion rule. For example, the Learn++.NSE variant (Muhlbaier and Polikar, 2007) relies on weighted majority voting for fusion, and keeps track of the performance of each classifier of the ensemble w.r.t. past batches of operational data. These measurements are used as voting weights, and are updated to integrate new batches of data, giving more weight to recent measurements. When a recurring concept is re-encountered, historical measures enable to detect the presence of a known concept, and increase the weights of related classifiers. Similarly the Fast Adapting Ensemble method (Ortíz Díaz *et al.*, 2015) implements heuristics to either activate or deactivate classifiers depending on the detected concept, as only activated classifiers participate in the final decision. When the presence of a previously encountered concept is detected in the operational data, classifiers associated to this concept re-activated, and their weights are adjusted.

Despite their regular update, the methods described above still apply a static selection or fusion rule (Britto *et al.*, 2014). The parameters are updated a posteriori after an observation over a window of past data, and remain static until the next update. Other methods have been proposed to provide a dynamic adaptation of the fusion rule, such the Mixture of Experts system (Jacobs *et al.*, 1991). It is comprised of an ensemble of neural network classifiers, as well as an additional gating network. For each input, the gating network is trained to compute the probabilities that each neural network of the ensemble is the most competent to classify it. These probabilities are then used to compute the ensemble final output, as the weighted average of the network outputs. While they provide a dynamic adaptation for each input pattern, the architecture of such methods remain static, as the gating network has to be updated with new reference data to remain relevant. It may also require the storage of previous data, for example to adapt its structure to the addition of a new classifier in the ensemble.

To address this limitation, dynamic selection methods have been proposed in the literature (Britto *et al.*, 2014). These involve, for each input to classify, the computation of a region Ψ defined as its k nearest neighbors in a set of validation data of known labels. Numerous methods have been proposed to compute classifier competence from the region Ψ. For example, in (Woods *et al.*, 1996), the accuracy of each classifier is computed as the percentage of correctly classified samples from Ψ, and the classifier with the highest accuracy is selected for classification. Other methods propose to select optimal ensemble subset instead of a single best classifier, such as the *DS-KNN* method (Santana *et al.*, 2006). It considers both accuracy and ensemble diversity measures. The N' most accurate classifiers in Ψ are first selected to generate an intermediate ensemble. Then, only the N'' most diverse classifiers of this ensemble are selected for classification, using double-fault diversity measures.

1.5.3 Challenges

In a VS environment, a FR system should be able to provide two levels of adaptation:

a. Adaptation of its training strategy. To adapt to newly available reference data, and refine its facial models by assimilating new concepts without corrupting past knowledge.

b. Adaptation of its operational behavior. In a video sequence, multiple concepts can be represented as the capture conditions evolve (e.g. movement of the individual). The system should be able to dynamically adapt its prediction behavior to always benefit from the most relevant knowledge to each capture, and avoid corrupting the prediction process with irrelevant knowledge.

Adaptive EoC methods can be used in a VS environment to fulfill these requirements. The first level may be achieved by training new classifiers for significantly different reference data, and the second should be addressed with a dynamic fusion rule, that only selects relevant classifiers among a diverse pool.

However, the compromise between accuracy and complexity must be addressed for the system to remain able to perform live recognition. While growing an EoC for each newly available batch of reference data will increase system complexity over time, only adding a new classifier when an abrupt change is detected might result in incomplete concept representations. In a VS environment, newly available data may also be used to refine knowledge about previously observed-concepts.

In the same way, the performance of dynamic selection methods depends heavily on the storage of a representative set of validation data, that is likely to grow over time as new concepts are detected in reference data. The estimation of competence regions involve a nearest neighbor estimation for each input capture, which computational complexity grows with the validation set. In addition, although a dynamic computation of competence regions enables to benefit from the most relevant information to each input ROI, these methods remain sensitive to the presence of unknown concepts in the operational streams. When presented with captures originating from concepts not represented in facial models nor validation data, incorrect competence prediction is likely to occur, either because of ill-defined competence regions (comprised of data from unrelated concepts), or poor classifier performance. In FRiVS, a dynamic adaptation of the fusion rule shouldn't interfere with the ability to perform live recognition, and corrupt system performance when unknown concepts are observed during operations.

In this thesis, an hybrid strategy involving both incremental learning and ensemble techniques is considered to address the *plasticity-stability* dilemma. On one hand, classifiers are updated through incremental learning to update facial models when gradual changes are observed, to refine knowledge about previously-observed concept. On the other hand, to assimilate new concepts without corrupting past knowledge, new classifiers are added to the ensembles when changes are detected. In addition, to provide a dynamic adaptation of the ensembles' fusion rules, classifier competence is estimated from concepts models used for *concept change* detection in reference videos, with represents a lower computational complexity than standard dynamic selection methods involving neighbor estimation.

CHAPTER 2

CONTEXT-SENSITIVE SELF-UPDATING FOR ADAPTIVE FACE RECOGNITION

Christophe Pagano[1], Eric Granger[1], Robert Sabourin[1], Pierluigi Tuveri[2], Gian Luca Marcialis[2], Fabio Roli[2]

[1] Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle, École de Technologie Supérieure, Université du Québec, Montréal, Canada

[2] Patter Recognition and Applications Group, Department of Electrical and Electronic Engineering, University of Cagliary, Italy

Book Chapter published in « Adaptive Biometric Systems: Recent Advances and Issues » by Springer, in 2015.

Abstract

Performance of state-of-the-art face recognition (FR) systems is known to be significantly affected by variations in facial appearance, caused mainly by changes in capture conditions and physiology. While individuals are often enrolled to a FR system using a limited number of reference face captures, adapting facial models through re-enrollment, or through self-updating with highly confident operational captures, has been shown to maintain or improve performance. However, frequent re-enrollment and updating can become very costly, and facial models may be corrupted if misclassified face captures are used for self-updating. This chapter presents an overview of adaptive FR systems that perform self-updating of facial models using operational (unlabelled) data. Adaptive template matching systems are first revised, with a particular focus on system complexity control using template management techniques. A new *context-sensitive* self-updating approach is proposed to self-update only when highly-confident operational data depict new capture conditions. This allows to enhance the modelling of intra-class variations while mitigating the growth of the system by filtering out redundant information, thus reducing the need to use costly template management techniques during operations. A particular implementation is proposed, where highly-confident templates are added

according to variations in illumination conditions detected using a global luminance distortion measures. Experimental results using three publicly-available FR databases indicate that this approach enables to maintain a level of classification performance comparable to standard self-updating template matching systems, while significantly reducing the memory and computational complexity over time.

2.1 Introduction

Automated face recognition (FR) has become an important function in a wide range of security and surveillance applications, involving computer networks, smart-phones, tablets, IP cameras, etc. Capturing faces in still images or videos allows to perform non-intrusive authentication in applications where the user's cooperation is either impossible (video-surveillance in crowded environments) or to be limited (continuous authentication). For example, in the context of controlled access to critical information on computer network systems, the face modality may allow for a continuous, non-intrusive authentication (Niinuma *et al.*, 2010). After initial log-in, a FR system may enroll the authenticated user using facial images captured from the computer's built-in camera, and design a facial model[1]. The user's identity may then be periodically validated using facial images captured over time without requiring active co-operation (i.e. password prompt).

However, limited user co-operation as well as uncontrolled observation environments often make FR a challenging task. It is well known that the performance of state-of-the-art FR systems may be severely affected by changes in capture conditions (e.g., variation in illumination, pose and scale), as well as individual physiology (Pagano *et al.*, 2014; De-la Torre *et al.*, 2014). Moreover, such systems are usually initialized with a limited number of high-quality reference face captures, which may generate non-representative facial models (not modelling all possible variations) (Pagano *et al.*, 2012).

[1] Depending on the classification system, a facial model may be defined as either a set of one or more reference face captures (template matching), or a statistical model estimated from reference captures (statistical classification).

To account for such intra-class variations, several solutions have been investigated in the literature over the past decade. They can be organized into the following two categories:

a. Development of discriminative features that are robust to environmental changes (De Marsico *et al.*, 2012; Wright *et al.*, 2009). These techniques usually aim to develop facial descriptors insensitive to changes in capture conditions, to mitigate their effects on the recognition process.

b. Storage (or synthetic generation) of multiple reference images to cover the different capture conditions that could be encountered during operations (Jafri and Arabnia, 2009; Li and Jain, 2011).

However, these approaches assume that FR is a stationary process, as they only rely on information available during enrolment sessions. In addition, depending on the application, a single enrolment session is often considered as multiple ones are not always possible (Rattani, 2010). This prevents to integrate new concepts[2] that may emerge during operations as capture conditions and individuals physiology evolve over time (for example due to natural lighting conditions and ageing).

To address this limitation, adaptive biometric systems have been proposed in the literature (Roli *et al.*, 2008), inspired by semi-supervised learning techniques for pattern recognition (Nagy, 2004). These systems are able to adapt facial models (sets of templates or classifier parameters) by exploiting (either on-line or off-line) faces captured during system operations. Common approaches in adaptive biometrics fall under *self-updating* and *co-updating*, depending on whether they rely on a single or multiple modalities. They usually either: 1) add novel captures to individual specific galleries (Roli and Marcialis, 2006), or 2), fuse new input data into common templates referred to as *super-templates*, containing all information (Jiang and Ser, 2002; Ryu *et al.*, 2006) for each modality (for example, virtual facial captures constructed with patches from operational data).

[2] A *concept* can be defined as the underlying data distribution of the problem under specific operating conditions (Narasimhamurthy and Kuncheva, 2007).

This chapter focuses on *self-updating* techniques with template matching systems for FR. These methods update template galleries using faces captured during operations that are considered highly-confident, i.e. that produce very high matching scores (surpassing a self-updating threshold) (Rattani *et al.*, 2009). Advantages and drawbacks of self-updating have been widely investigated (Marcialis *et al.*, 2008; Rattani *et al.*, 2009). While these methods have been show to significantly improve the performance of biometric systems over time, an updating strategy only relying on matching score values may add redundant template to the galleries. This can significantly increase system complexity over time with information that do not necessarily improve performance, and also eventually reduce its response time during operations. To bound this complexity, template management methods (e.g. pruning) have been proposed in literature (Freni *et al.*, 2008; Marcialis *et al.*, 2008; Rattani *et al.*, 2009). While clustering-based methods showed the most promising results, they remain computationally complex and thus not suited for seamless operations, if self-updating is performed frequently.

In this chapter, a survey of state-of-the-art techniques for adaptive FR using self-updating is presented, along with the key challenges facing these systems. An experimental protocol involving three real-life facial datasets (DIEE (Rattani *et al.*, 2013), FIA (Goh *et al.*, 2005) and FRGC (Phillips *et al.*, 2005)) is proposed to evaluate the benefits and drawbacks of a self-updating methodology applied to a template matching system, with a particular focus on the management of system complexity. To address this challenge, a *context-sensitive* self-updating technique is proposed for template matching systems, combining a standard self-updating procedure and a change detection module. With this technique, only operational faces that were captured under different capture conditions are added to an individual's template gallery. More precisely, the addition of a new capture into the galleries depends on two conditions: 1) it's matching score is above the self-updating threshold (highly confident capture), and 2), the capture contains new information w.r.t. the samples already present in the gallery (i.e. captured under different conditions). This strategy allows to benefit from contextual information available in operational captures to limit the growth in system complexity. With this technique, one can avoid frequent uses of costly template management schemes, while still enhancing

intra-class variation in facial models with relevant templates. A particular implementation of this proposed technique is considered for a basic template matching system, where changes are detected in illumination conditions.

The rest of this chapter is organized as follows. Section 2 provides a general survey of self-updating algorithms in the context of adaptive biometric systems. Then, section 3 introduces the new *context-sensitive* self-updating technique based on the detection of changes in capture conditions, and Section 4 presents the proposed experimental methodology. Finally, experimental results are presented and discussed in Section 5.

2.2 Self-Updating for Face Recognition

2.2.1 A General System

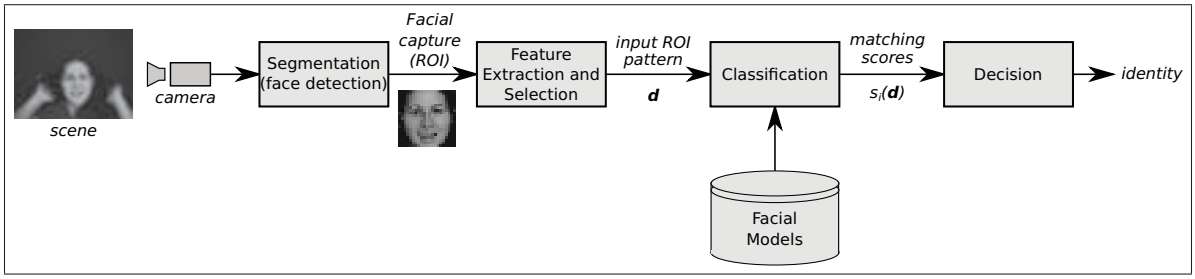


Figure 2.1 General FR system trained for N individuals.

Figure 2.1 presents a generic system for the recognition of faces in images (stills or video frames) captured from a camera. It is composed by four modules: segmentation, feature extraction, classification and decision. In addition, facial models of the *N* enrolled individuals are stored into the system, to be used by the classification module to produce matching scores for each individual.

During operations, faces are isolated in the image using the segmentation module, which produces the regions of interest (ROIs). Then, discriminant features are extracted from each ROI (e.g. eigenfaces (Turk and Pentland, 1991) of local binary patterns (Ahonen *et al.*, 2006)) to

produce the corresponding pattern $\mathbf{d} = (d[1], ..., d[F])$ (with F the dimensionality of the feature space). This pattern is then compared to the facial model of each enrolled individual i by the classifier, which produces the corresponding matching scores $s_i(\mathbf{d})$, $(i = 1, ..., N)$.

The facial models are usually designed a priori using one or several reference patterns, from which the same features have been extracted, and their nature depends on the type of classifier used in the system. For example, with a template matcher, a facial model of an individual i can be a gallery of one or several reference patterns $\mathbf{r}_{i,j}$ $(j = 1, ..., J)$, in which case matching scores for each operational pattern $\mathbf{d}$ would be computed from distance measures to these patterns. Classification may also be performed using neural networks (e.g multi-layer perceptrons (Riedmiller, 1994) and ARTMAP neural networks (Carpenter *et al.*, 1991)) or statistical classifiers (e.g. naïve Bayes classification (Duda and Hart, 1973)), in which case the facial models would consist of parameters estimated during their training using the reference patterns $\mathbf{r}_{i,j}$ (e.g. neural networks weights, statistical distribution parameters, etc.).

Finally, the decision module produces a final response according to the application. For example, an identification system for surveillance may predict the identity of the observed individual with a maximum rule, selecting the enrolled individual with the highest matching score, while a verification system for access control usually confirms the claimed identity by comparing the corresponding matching score to a decision threshold

2.2.2 Adaptive Biometrics

As mentioned earlier, the performance of FR systems can be severely affected by changes in capture conditions. Intra-class variations can be observed in the input data, as a consequence of changes in capture conditions (scene illumination, facial pose angle w.r.t. the camera, etc.) or individuals physiology (facial hair, ageing, etc.). Such diversity is difficult to represent using the limited amount of reference captures used for initial facial model design. To address this limitation, adaptive biometric systems have been proposed in the literature, providing the

option for continuous adaptation of the facial models using the operational data (Rattani, 2010; Rattani *et al.*, 2009).

Adaptation can be either supervised or unsupervised, depending on the labelling process of the operational data. In semi-supervised learning (Zhu, 2005), the facial model of each individual enrolled to the system is updated using operational data labelled as the same individual by the classification system. For example, a gallery $\mathscr{G}_i$ of reference patterns may be augmented with highly-confident operational input patterns $\mathbf{d}$ matched to the facial model of individual i. While this enables to refine facial models, the performance of such systems is strongly dependent on their initial classification performance. In addition, the integration of mislabelled captures could corrupt facial models, thus affecting the accuracy of the system for the corresponding individuals (Rattani *et al.*, 2009, 2013)

An adaptive biometric system can also perform supervised adaptation, where the operating samples used to update the system are manually labelled, or obtained through some re-enrolment process (Rattani *et al.*, 2009). While supervised adaptation may represent an ideal scenario with an error-free labelling process, human intervention is often costly or not feasible. Depending on the application, the ability to perform semi-supervised adaptation may be the only viable solution, which has lead to the development of various strategies to increase the robustness of such systems.

These techniques can be categorized as *self-update* (Jiang and Ser, 2002; Ryu *et al.*, 2006) and *co-update* techniques (Rattani *et al.*, 2009, 2008), depending on whether a single or multiple modalities are considered for the update of facial models with highly-confident patterns. This chapter focuses on *self-updating* methods for FR, where facial models are defined by galleries of reference patterns.

2.2.3 Self-Updating Methods

In the context of FR systems, self-updating methods update the facial models using only highly-confident operational captures, i.e. with matching scores surpassing a very high threshold, to prevent possible corruptions due to misclassification.

2.2.3.1 General Presentation

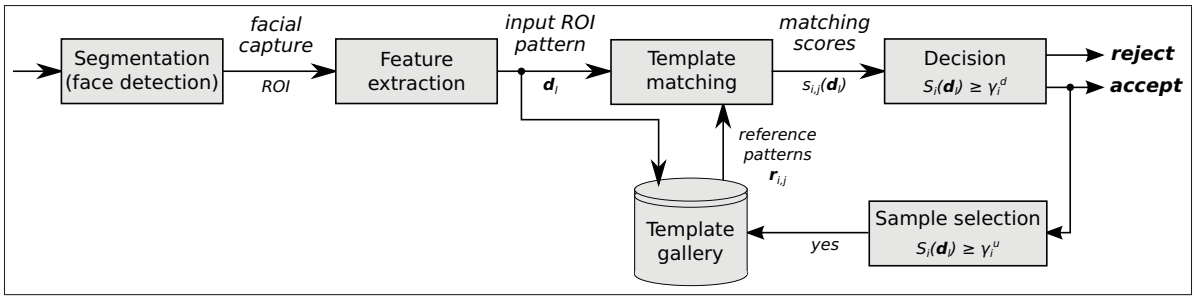


Figure 2.2 A FR system based on template matching that allows for self-update.

To illustrate this principle, it is applied to a template matching system, presented in Figure 2.2. In this system, inspired by (Rattani *et al.*, 2011), the facial model of each individual i is designed by storing initial reference patterns from a labelled dataset into a gallery $\mathscr{G}_i = \{\mathbf{r}_{i,1}, \mathbf{r}_{i,2}, ...\}$ (in this case, the terms *pattern* and *template* are used indiscriminately). To simplify the notation, the remaining of this section will omit the subscript i and only consider one individual, as this methodology can be extended to many with individual specific galleries and thresholds.

Alg. 2.1 presents a generic algorithm for self-updating a template gallery $\mathscr{G}$ with several reference patterns $\mathbf{r}_j$ $(j = 1, ..., J)$. During operations, the system is presented with an unlabelled data set $\mathscr{D}$ of L facial captures. For each sample $\mathbf{d}_l$, similarity measures to each reference $\mathbf{r}_j$ in the galley are used to compute the set of matching scores $s_j(\mathbf{d}_l)$ $(j = 1, ..., J)$. Then, the final score $S(\mathbf{d}_l)$ is computed as a combination of $s_j(\mathbf{d}_l)$ (e.g. the maximum fusion rule), and positive prediction is output if it surpasses the decision threshold γ^d. Finally, the sample selection

Algorithm 2.1: Self-update algorithm for adapting template galleries.

Input: *Gallery with initial templates* $\mathscr{G} = \{\boldsymbol{r}_1, ..., \boldsymbol{r}_J\}$, *unlabeled adaptation set* $\mathscr{D} = \{\boldsymbol{d}_1, ..., \boldsymbol{d}_L\}$;
Output: *Updated Gallery* $\mathscr{G}' = \{\boldsymbol{r}_1, ..., \boldsymbol{r}_{J'}\}$, $J' \geq J$;
- Estimate updating threshold $\gamma^u \geq \gamma^d$ from $\mathscr{G}$;
- $\mathscr{G} \leftarrow \mathscr{G}'$; `//Initialization with previous state`
for *all samples* $d_l \in \mathscr{D}$ $(l = 1, ..., L)$ **do**
    `//Prediction score for each reference`
    for *all references* $r_j \in \mathscr{G}$ $(j = 1, ..., J)$ **do**
        - $s_j(\mathbf{d}_l) \leftarrow similarity\_measure(\mathbf{d}_l, \mathbf{r}_j)$;
    end
end
- $S(\mathbf{d}_l) \leftarrow \max_{j \in [1,J]} \{s_j(\mathbf{d}_l)\}$; `//Maximum fusion of scores`
if $S(\boldsymbol{d}_l) \geq \gamma_d$ **then**
    - Output positive prediction ;
    `//Add the sample which similarity surpasses` γ^u `to the gallery`
    if $S(\boldsymbol{d}_l) \geq \gamma^u$ **then**
        - $\mathscr{G}' \leftarrow \mathscr{G}' \cup \mathbf{d}_l$;
    end
end

module relies on a stricter updating threshold γ^u (usually $\gamma^u \geq \gamma^d$), updating the gallery $\mathscr{G}$ with $\mathbf{d}_l$ if $S(\mathbf{d}_l) \geq \gamma^u$, i.e. if the prediction has a high degree of confidence.

2.2.3.2 Challenges

While self-updating methods have been shown to improve system accuracy over time, the adaptation of the facial models using operational data might be detrimental, and the selection of the updating threshold is critical (Rattani *et al.*, 2009). To prevent a decline in classification performance, the use of a strict updating threshold may enable to reduce the probability of updating the facial models with misclassified patterns (Liu *et al.*, 2003; Roli and Marcialis, 2006; Ryu *et al.*, 2006). However, it has been argued that updating with only highly confident patterns may result in the addition of redundant information in the galleries, and thus a marginal

gain in performance at the expense of a considerable increase in system complexity (Rattani *et al.*, 2009).

In addition, operational samples with more drastic changes are less likely to generate classification scores surpassing the updating threshold, preventing the classification system to assimilate this new information. To address this limitation, *co-updating* methods have been proposed to benefit from complementary biometric systems (Rattani *et al.*, 2009, 2008). Each system is initialized with reference templates from a different source (or different features extracted from the same source), and performs classification of operational input data. In the same way as self-updating techniques, each system selects highly-confident samples based on an updating threshold, but this information is also shared with other systems. If the classification score of one system surpasses its updating threshold, the others will also consider the corresponding samples as highly confident, and perform adaptation. This enables to increase to probability of updating with different but genuine operational data, by relying on the supposition that a drastic change on one source is not necessarily observed on others. A recent model has been proposed to estimate optimal amounts of samples and iterations to improve system's performance under specific updating constraints (Didaci *et al.*, 2014). This model has shown to be effective under the stringent hypothesis of 0% false alarm rate for the updating threshold of both systems. While *co-updating* is usually applied with multiple biometric traits, it could also be applied in, for example, a FR scenario involving multiple cameras. In this situation, relying on multiple point of views could mitigate the effect of disruptions such as motion blur that would be less likely to affect every camera at the same time.

Finally, system complexity is a critical issue for template matching systems in live FR. The ability to operate seamlessly depends on the computational complexity of the recognition operation, which is usually directly related to gallery sizes. Several template management strategies have been proposed to limit complexity in self-updating systems. In (Freni *et al.*, 2008), template replacement strategies have been experimented to perform self-update in a constrained environment, where the maximum number of templates in a gallery is fixed by the user. When the maximum size is reached, several criteria have been experimented to determine which ob-

solete template can be replaced, such as FIFO, LFU and clustering algorithms. Among them, the clustering algorithm MDIST showed the most promising results, reducing the number of impostors samples by maintaining a gallery with very close samples. While these methods enable to compromise between system performance and complexity, they remain computationally costly, and may interfere with seamless long-term operations. Once the maximum gallery size is reached, such process would have to be performed for each new highly-confident template, thus increasing system response time. To reduce these occurrences, operational data containing redundant information should be filtered out during operations. This would limit the self-updating process to only operational templates with relevant information, i.e. templates improving intra-class variability in facial models.

2.3 Self-Updating Driven by Capture Conditions

This chapter introduces a new self-updating method that efficiently self-updates facial models based on capture conditions. This methodology is illustrated using a template matching system performing self-updating, as presented in (Rattani *et al.*, 2011). As discussed in the previous sections, such methodology can significantly improve the overall classification performance through a better modelling of intra-class variations, specifically in applications exhibiting significant variations in capture conditions (e.g. continuous authentication using webcams). However, updating the galleries with only highly-confident inputs may not always provide new and beneficial information, as those samples are usually well-classified by the system, which could lead to an unnecessary increase in system complexity (e.g. the number of reference patterns stored in the galleries) (Rattani *et al.*, 2009). While this complexity can be mitigated with template-management techniques (Freni *et al.*, 2008), frequent gallery filtering may interfere with seamless operations over time.

To address this limitation, this section proposes a *context-sensitive* self-updating technique that integrates a template filtering process during operations. It is designed to ensure that only highly-confident data captured under novel conditions are added to template galleries, thus limiting the growth in memory complexity with redundant samples. In fact, in FR, intra-

class variations in facial appearance are often related to changes in capture conditions (e.g. environmental illumination, facial pose, etc.) (Pagano *et al.*, 2014; De-la Torre *et al.*, 2014), and such information can be detected during operations. Following this intuition, when a highly confident ROI pattern surpasses the updating threshold, non-discriminative information related to capture conditions are extracted to evaluate whether it has been captured under different conditions that of the reference templates already stored in the gallery. If not, the pattern is discarded, and the gallery is not augmented.

2.3.1 Framework For Context-Sensitive Self-Update

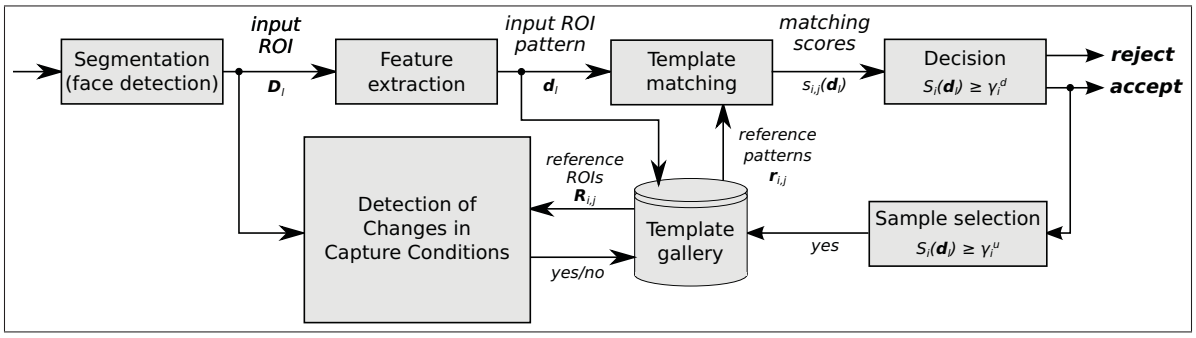


Figure 2.3 A template matching system that integrates context-sensitive self-updating.

The diagram of a general template-matching system that employs the new *context-sensitive* technique is presented in Figure 2.3. It augments the system presented in Figure 2.2 with an additional decision module to detect changes in capture conditions.

In the same way than standard self-updating systems, when presented with a unlabelled data set $\mathscr{D} = \{\mathbf{d}_1, ..., \mathbf{d}_L\}$, this system first selects highly-confident samples to perform adaptation of the template gallery $\mathscr{G}_i$, i.e. the set $\mathscr{D}' = \{\mathbf{d}_{l'} | S_i(\mathbf{d}_{l'}) \geq \gamma_i^u\}$. Then, an additional test is performed on these samples, only to select a final subset captured under novel capture conditions. To extract additional non-discriminative information, the individual galleries are augmented with the input ROIs $\mathbf{R}_{i,j}$ from which the reference patterns $\mathbf{r}_{i,j}$ are extracted. The augmented galleries are stored as $\mathscr{G}_i = \{\{\mathbf{R}_{i,1}, \mathbf{r}_{i,1}\}, \{\mathbf{R}_{i,2}, \mathbf{r}_{i,2}\}, ...\}$. This additional measurement enables

to maximize the intra-class variation of the galleries while mitigating their growth by rejecting redundant information. For example, contextual information such as environmental illumination or facial pose w.r.t. the camera can be measured on ROIs, to be compared with ROIs in the galleries.

2.3.2 A Specific Implementation

As a basic example of the framework presented in Figure 2.3, a particular implementation is proposed. It relies on the detection of changes in illumination conditions.

2.3.2.1 A Template Matching System

For classification, a standard template matching system is considered. For each individual i, a dedicated facial model is stored as a template gallery $\mathscr{G}_i = \{\{\mathbf{R}_{i,1}, \mathbf{r}_{i,1}\}, \{\mathbf{R}_{i,2}, \mathbf{r}_{i,2}\}, ..., \{\mathbf{R}_{i,J_i}, \mathbf{r}_{i,J_i}\}\}$, as well as user-specific decision γ_i^d and updating γ_i^u thresholds.

For each input ROI isolated through segmentation, the corresponding pattern $\mathbf{d}_l$ is extracted using a Multi-Bloc Local Binary Pattern (LBP) (Ahonen *et al.*, 2006) algorithm. Features for block sizes of 3x3, 5x5 and 9x9 pixels are computed and concatenated with the grayscale pixel intensity values, and PCA is used to reduce the dimensionality to $F = 32$[3]. The matching score for each individual i is then computed following:

$$S_i(\mathbf{d}_l) = \frac{1}{J_i} \cdot \sum_{j=1}^{J_i} s_{i,j}(\mathbf{d}_l) = \frac{1}{J_i} \sum_{j=1}^{J_i} \frac{\left[\sqrt{F} - d_{Eucl}(\mathbf{d}_l, \mathbf{r}_{i,j})\right]}{\sqrt{F}} \tag{2.1}$$

where $d_{Eucl}(\mathbf{d}_l, \mathbf{r}_{i,j})$ is the Euclidean distance between input pattern $\mathbf{d}_l$ and template $\mathbf{r}_{i,j}$ (with $j = 1, ..., J_i$) and J_i the total number of templates in $\mathscr{G}_i$. The matching scores $s_{i,j}(\mathbf{d}_l)$ are here computed as the normalized opposite to the distance $d_{Eucl}(\mathbf{d}_l, \mathbf{r}_{i,j})$ (a score of 1 is achieved for a null distance). The final matching score $S_i(\mathbf{d}_l)$ is obtained from the combination of these scores using the average fusion rule.

[3] This value has been determined experimentally as an optimal trade-off between accuracy and computational complexity using a nearest-neighbour classifier with Euclidean distance.

Finally, the system outputs a positive prediction for individual i if $S_i(\mathbf{d}_l) \geq \gamma_i^d$, and selects $\mathbf{d}_l$ as a highly confident face capture for individual i if $S_i(\mathbf{d}_l) \geq \gamma_i^\mu$.

2.3.2.2 Detecting Changes in Capture Conditions

In Figure 2.3, for each individual i, the input ROIs $\mathbf{D}_l$ corresponding to highly-confident operational captures are compared to the reference ROIs $\mathbf{R}_{i,j}$ $(j = 1,...,J_i)$ stored in the galleries, to asses whether the capture conditions are novel enough to justify an increase in complexity. The universal image quality index Q (Wang and Bovik, 2002) is considered to measure the distortion between $\mathbf{D}_l$ and each reference ROI $\mathbf{R}_{i,j}$. This measure is a particular case of the Structural Similarity Index Measure (SSIM) presented in (Wang *et al.*, 2004). It can be written as a product of the three factors – loss of correlation, luminance distortion and contrast distortion:

$$Q(\mathbf{R}_{i,j},\mathbf{D}_l) = \frac{\sigma_{\mathbf{R}_{i,j},\mathbf{D}_l}}{\sigma_{\mathbf{R}_{i,j}} \cdot \sigma_{\mathbf{D}_l}} \cdot \frac{2\bar{\mathbf{R}}_{i,j} \cdot \bar{\mathbf{D}}_l}{\bar{\mathbf{R}}_{i,j}^2 + \bar{\mathbf{D}}_l^2} \cdot \frac{2\sigma_{\mathbf{R}_{i,j}} \cdot \sigma_{\mathbf{D}_l}}{\sigma_{\mathbf{R}_{i,j}}^2 + \sigma_{\mathbf{D}_l}^2} \tag{2.2}$$

where $\bar{\mathbf{R}}_{i,j}$ and $\bar{\mathbf{D}}_l$ are the average images, $\sigma_{\mathbf{R}_{i,j}}$ and $\sigma_{\mathbf{D}_l}$ their variances, and $\sigma_{\mathbf{R}_{i,j},\mathbf{D}_l}$ the covariance.

To accommodate spatial variations in image distortion, statistical features for Eq. 2.2 may be measured locally. A local quality index $Q(\mathbf{R}_{i,j}[k],\mathbf{D}_l[k])$ is thereby calculated, where $\mathbf{D}_l[k]$ ($\mathbf{R}_{i,j}[k]$) corresponds to window of $\mathbf{D}_l$ ($\mathbf{R}_{i,j}$) sliding from the top-left corner to the bottom right corner for a total of K steps. These local measurements can then be combined into the global quality index GQ following:

$$GQ(\mathbf{R}_{i,j},\mathbf{D}_l) = \frac{1}{K}\sum_{k=1}^{K} Q(\mathbf{R}_{i,j}[k],\mathbf{D}_l[k]) \tag{2.3}$$

In this chapter, the proposed template filtering strategy is implemented through a detection of changes in ROI illumination conditions only. For that intent, the second term of the quality

index Q (see Eq. 2.2) is considered, to compute the global luminance quality (GLQ) following:

$$GLQ(\mathbf{D}_l, \mathbf{R}_{i,j}) = \frac{1}{K}\sum_{k=1}^{K} LQ(\mathbf{R}_{i,j}[k], \mathbf{D}_l[k]) = \frac{1}{K}\sum_{k=1}^{K} \frac{2.\bar{\mathbf{D}}_l[k]\ .\ \bar{\mathbf{R}}_{i,j}[k]}{\bar{\mathbf{D}}_l[k]^2 + \bar{\mathbf{R}}_{i,j}[k]^2} \tag{2.4}$$

where the local luminance quality measurements LQ measure the proximity of the average luminance between each window. Highly confident captures $\mathbf{D}_l$ are then used to update the gallery $\mathscr{G}_i$ if and only if

$$\frac{1}{J_i}\sum_{j=1}^{J_i} GLQ(\mathbf{D}_l, \mathbf{R}_{i,j}) \geq \gamma_i^c \tag{2.5}$$

with γ_i^c the capture condition threshold, computed as the average GLQ between all the references captures in $\mathscr{G}_i$.

2.4 Simulation Methodology

This section presents several experimental scenarios involving three real-world FR databases. The proposed simulations emulate realistic FR applications of different orders of complexity, with variations in capture conditions. The objective is to observe and compare the performance of new and reference self-updating techniques under different operation conditions, and within a basic template matching system described in Section 2.3.2.

2.4.1 Face Recognition Databases

Three publicly-available FR databases are considered for simulation. To standardize the experimental protocol, each database is separated into 6 different batches for all individuals. These scenarios are summarized at the end of Section 2.4.1, in Table 2.1.

2.4.1.1 Multi-Modal Dipartimento di Ingegneria Elettrica ed Elettronica

The multi-modal Dipartimento di Ingegneria Elettrica ed Elettronica[4] (DIEE) dataset (Rattani *et al.*, 2013) regroups face and fingerprint captures of 49 individuals. In this study, only facial

[4] Department of Electrical and Electronic Engineering.

captures are considered. For each individual, 60 facial captures have been acquired over 6 sessions at least three weeks apart, with 10 captures per session. The collection process spaned over a period of 1.5 years.

For simulations, the facial captures or each individuals are separated into 6 batches corresponding to the capture sessions. ROIs have been extracted with a semi-manual process (Marcialis *et al.*, 2014): an operator first selected the eyes in each frame, and the cropped region was then determined as the square of size $2d * 2d$ (d being the distance between the eyes), with the eyes located at the position $(d/2, d/4)$ and $(3 \cdot d/2, d/4)$. In this process, faces have been rotated to align the eyes to minimize intra-class variations (Gorodnichy, 2005a), and then normalized to a size of 70x70 pixels.

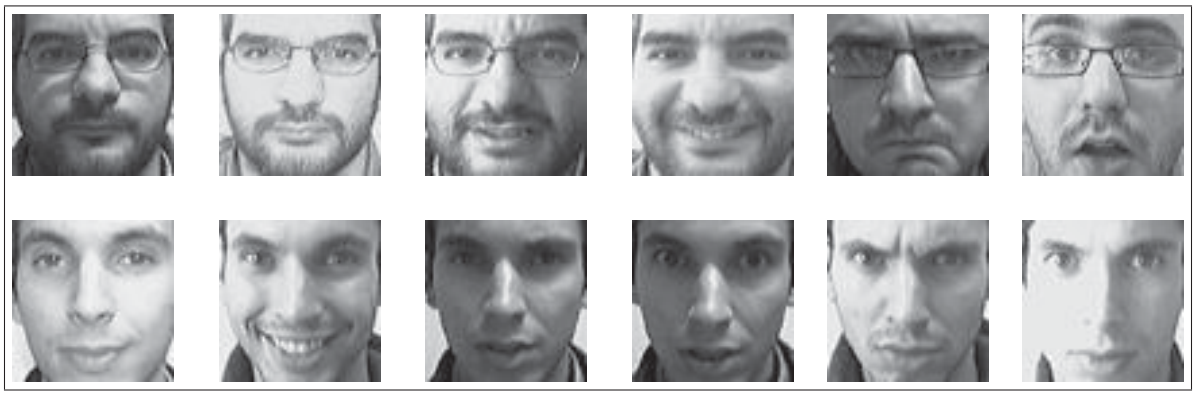


Figure 2.4 DIEE dataset. An example of randomly chosen facial captures for two individuals.

This dataset was explicitly collected to evaluate the performance of self-update and co-training algorithms. Over the 6 sessions, gradual changes can be observed in facial pose, orientation, and illumination (see examples in Fig. 2.4). While these changes generate visible differences in facial captures, the position of the individuals and their distance to the camera are controlled. For this reason, this dataset represents the easiest problem in this study, simulating an application of continuous authentication of individuals over a computer network.

2.4.1.2 CMU Faces in Action

The Carnegie Mellon University Faces In Action (FIA) dataset (Goh *et al.*, 2005) contains a set of 20-second videos for 221 participants, mimicking a passport checking scenario in both indoor and outdoor environments. Videos have been captured in three separate sessions of 20 seconds at least one month apart, with 6 Dragonfly Sony ICX424 cameras (640x480 pixel resolution, 30 images per second). Cameras were positioned at 0.83m of the subjects, mounted on carts at three different horizontal angles (0° and $\pm 72.6^\circ$), with two focal lengths (4 and 8mm) each.

Figure 2.5 FIA dataset. An example of randomly chosen facial captures for two individuals.

In this chapter, only ROIs captured during the indoor sessions, and using the frontal camera with 8mm focal length are considered. ROIs have been extracted using the OpenCV implementation of Viola-Jones face and eye detection algorithm (Viola and Jones, 2004). In the same way than with DIEE, faces have been rotated to align the eyes (Gorodnichy, 2005a), and normalized to a size of 70x70 pixels. For simulations, sequences from each session have been divided into two sub-sequences, in order to organize the facial captures into 6 batches.

This dataset simulates an open-set surveillance scenario as found in face re-identification applications. A restrained subset of 10 individuals of interest are monitored, but in an environment where a majority of ROIs are capture from non-target individuals. The 10 individuals of interest

enrolled to the systems have been chosen with two experimental constraints: 1) the individuals must be present in all capture sessions, and 2), at least 30 ROIs per session have been extracted by the face detection algorithm.

Faces in this data set have been captured in semi-controlled capture conditions, where the individuals entered the scene and walked to stop at the same distance from the cameras, and talked while moving their head with natural movements until the end of the session. In addition to variations in illumination and facial expressions, ROIs also incorporate variations in pose, resolution (scaling), motion blur and ageing.

2.4.1.3 Face Recognition Grand Challenge

The Face Recognition Grand Challenge (FRGC) dataset as been collected at University Notre Dame (Phillips *et al.*, 2005). In this chapter, the still face images of this dataset are considered. They were captured over an average of 16 sessions for 222 individuals for the training subset, and up to 22 sessions for the validation one, using a 4 Megapixels Canon camera. Each session contains four controlled and two uncontrolled captures, with significantly different illumination and expression.

Figure 2.6 FRGC dataset. An example of randomly chosen facial captures for two individuals

Overall, 187 individuals have been selected for experiments, for which more than 100 ROIs are available (around 133 in average). In the same way than with the other datasets, 6 batches of the the sane size have been created for each individual, respecting the temporal relation between the capture sessions. ROIs have been extracted in the same way than with the DIEE dataset (Marcialis *et al.*, 2014), using the position of the eyes already available in the FRGC dataset.

This dataset simulates a wide-range identification application, with multiple re-enrolment sessions where a very limited amount of reference templates are captured. Recurring and unpredictable changes in illumination and facial expression emerge in the operational environment in every capture session.

Table 2.1 Summary of the three experimental scenarios.

Dataset	Scenario	# enrolled individuals	# enrolment sessions	# ROIs per batch	Sources of variation
DIEE	continuous authentication	49	6	10	illumination, expression
FIA	video-surveillance	10	3	69	illumination, expression, pose, resolution, ageing, scaling, blur
FRGC	wide-range identification	187	16	22	illumination, expression, ageing

2.4.2 Protocol

The following three template matching systems are experimentally compared in this chapter:

a. **baseline system**, performing template matching in the same way as in Figure 2.3, but without any adaptation of the template galleries $\mathscr{G}_i$. User-specific decision thresholds γ_i^d are stored for decision.

b. standard **self-updating** system, updating the template galleries $\mathscr{G}_i$ with highly confident ROI patterns, which scores surpass user-specific updating thresholds γ_i^u, and decision thresholds γ_i^d.

c. proposed **context-sensitive self-updating** system, only updating the template galleries $\mathscr{G}_i$ with highly confident samples that also passed the concept change test (Eq. 2.5), using user-specific updating γ_i^u, capture condition γ_i^c and decision thresholds γ_i^d.

2.4.2.1 Simulation Scenario

The scenario descried below is considered for each database. At each time step $t = 1,...,6$, and for each individual $i = 1,...,N$, the performance of the baseline and the two self-updating systems updated with batch $b_i[t-1]$ is evaluated on batch $b_i[t]$. The self-updating systems are updated, and then tested with batch $b_i[t+1]$, and so on. A pseudo-code of the simulation process is presented in Alg. 2.2.

For each system, the individual galleries $\mathscr{G}_i$ are initialized with the two first samples of the corresponding initial batches $b_i[1]$. For *context-sensitive* self-updating, corresponding ROIs are also stored to compute GLQ measures during operations (see Eq. 2.4). Then, the initial values of the decision thresholds γ_i^d are computed using negative distribution estimation: each gallery $\mathscr{G}_i$ is compared to every other gallery to generate negative scores, and a threshold γ_i^d is chosen as the highest possible value respecting an operational false alarm constraint. For the self-updating variants, the updating threshold γ_i^u is initialized in the same way, and for the *context-sensitive* self-updating system, γ_i^c is computed as the average GLQ measure between each ROI in $\mathscr{G}_i$.

Then, for each system, performance is evaluated using the remaining patterns from $b_i[t]$ to compute genuine scores, and a random selection of impostor patterns for the impostor scores. For the DIEE and FRGC datasets, impostor patterns for each individual are randomly selected among batches from other individuals. In the case of the FIA dataset, impostor patterns are

Algorithm 2.2: Protocol for simulations.

for $i = 1, ..., N$ **do**
  //Initialization of the galleries $\mathscr{G}_i$ for each individual
  - $\mathscr{G}_i \leftarrow$ first 2 patterns of $b_i[1]$;
end
for $i = 1, ..., N$ **do**
  //Initialization of thresholds for each individual
  - Evaluate update and decision thresholds γ_i^u and γ_i^d using negative distribution estimation;
  - Initialize change detection threshold γ_i^c as the average GLQ measure between each ROI in $\mathscr{G}_i$;
end
for $i = 1, ..., N$ **do**
  //Processing of remaining samples from $b_i[1]$
  - Estimate genuine scores using remaining samples from $b_i[1]$;
  - Estimate impostor samples using a random selection of impostor samples;
  - Update gallery;
  - Update thresholds;
end
for $t = 2, ..., 6$ **do**
  // Remaining data blocks
  for $i = 1, ..., N$ **do**
    - Estimate genuine scores using remaining samples from $b_i[t]$;
    - Estimate impostor samples using a random selection of impostor samples;
    - Update gallery;
    - Update thresholds ;
  end
end

selected from the non-target dataset individuals during the same session. To avoid any bias in performance evaluation, the same amount of impostor and genuine patterns are considered.

Finally, using genuine and impostor patterns, the self-updating systems galleries are updated according to their updating strategies, and the thresholds are re-estimated using the same methodology. This scenario is then reproduced for the remaining 5 batches.

2.4.2.2 Performance Measures

For each system, performance is measured with average true positive rate (tpr) and false positive rate (fpr) for each individual. These are respectively the proportion of genuine patterns correctly classified over the total number of genuine patterns (tpr), and the proportion of impostor patterns classified as genuine over the total number of negative patterns (fpr). These measures depend on the decision thresholds γ_i^d, computed during update to respect a given fpr constraint.

System complexity is also presented, as the average number of templates in the galleries. In addition, facial model corruption due to the addition of misclassified templates in the galleries is presented as the ratio of impostor over genuine templates. Following Doddington's classification (Doddington *et al.*, 1998), only the 10 galleries with the highest ratio are presented, to focus on lamb-type individuals which are easy to imitate.

Finally, a constraint of fpr $= 5\%$ has been chosen to compute the decision thresholds γ_i^d. In addition, for each scenario, the updating thresholds γ_i^u correspond to an ideal fpr $= 0\%$ and a laxer fpr $= 1\%$. For each performance measure, results are presented as the average and standard deviation values for every enrolled individual, computed using a Student distribution and a confidence interval of 10%.

2.5 Simulation Results

2.5.1 Continuous User Authentication with DIEE Data

Figure 2.7 presents the average performance results of the baseline, self-updating and *context-sensitive* self-updating techniques within the template matching system described in Section 2.3.2. Results are presented for the ideal fpr $= 0\%$ updating thresholds for the self-updating techniques.

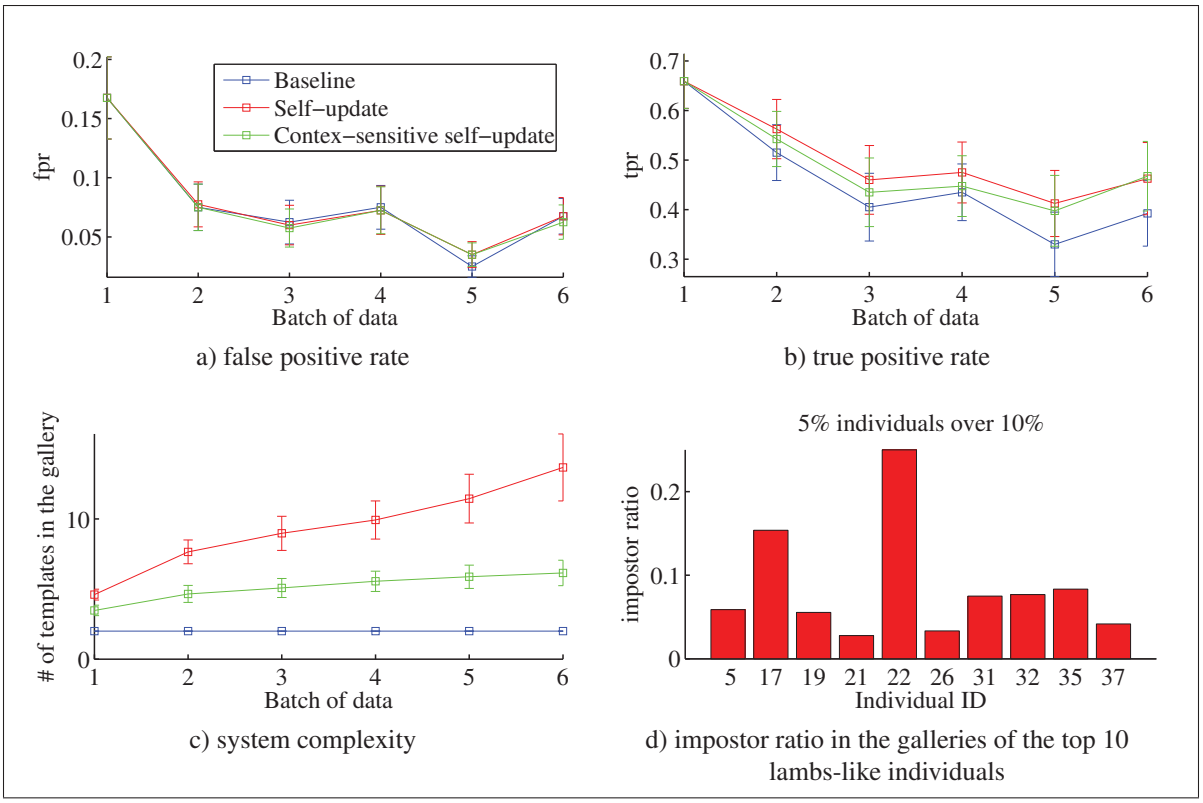


Figure 2.7 Simulation results with DIEE dataset where the updating threshold is selected for fpr $= 0\%$.

While all 3 systems present similar fpr (between $7-17\%$) in Fig. 2.7 (a), a significant differentiation can be observed in the tpr with batches 5 and 6 (Fig. 2.7 (b)). In fact, the introduction of batch 5 generates a decline in tpr performance for the baseline system (from $43.5\pm5.7\%$ down to $33.0\pm6.5\%$), that ends at tpr $= 39.3\pm6.6\%$ at batch 6. On the other hand, the self-updating and *context-sensitive* self-updating systems exhibit a moderate decline (respectively from $47.5\pm6.1\%$ to $41.3\pm6.7\%$), and end at a higher performance of tpr $= 46.3\pm7.3\%$.

Even with a fpr $= 0\%$ updating threshold, it can be observed that this FR scenario benefits from a self-updating strategy, as the addition of up to an average 13.7 ± 2.4 templates in the galleries (see Fig. 2.7 (c)) enabled to increase the system's performance. In addition, despite the limited amount of captures (10 per session), the filtering of the *context-sensitive* self-updating system enabled to maintain a comparable level of performance with a significantly lower amount of templates in the gallery, ending at an average of 6.1 ± 0.9 templates.

Despite the relative simplicity of this scenario and the restrictive updating threshold, impostor templates have been incorrectly added to the galleries during the updating process. Following Doddington's analysis, the ratio of impostor over genuine templates in the galleries of the top 10 lamb individuals (i.e. the individuals with the highest ratio) are presented in Fig. 2.7 (d). While 95% of the galleries contain under 10% of impostor samples, two lamb-like individuals (ID 17 and 22) stand out with over 10% and 20% impostor samples in their galleries.

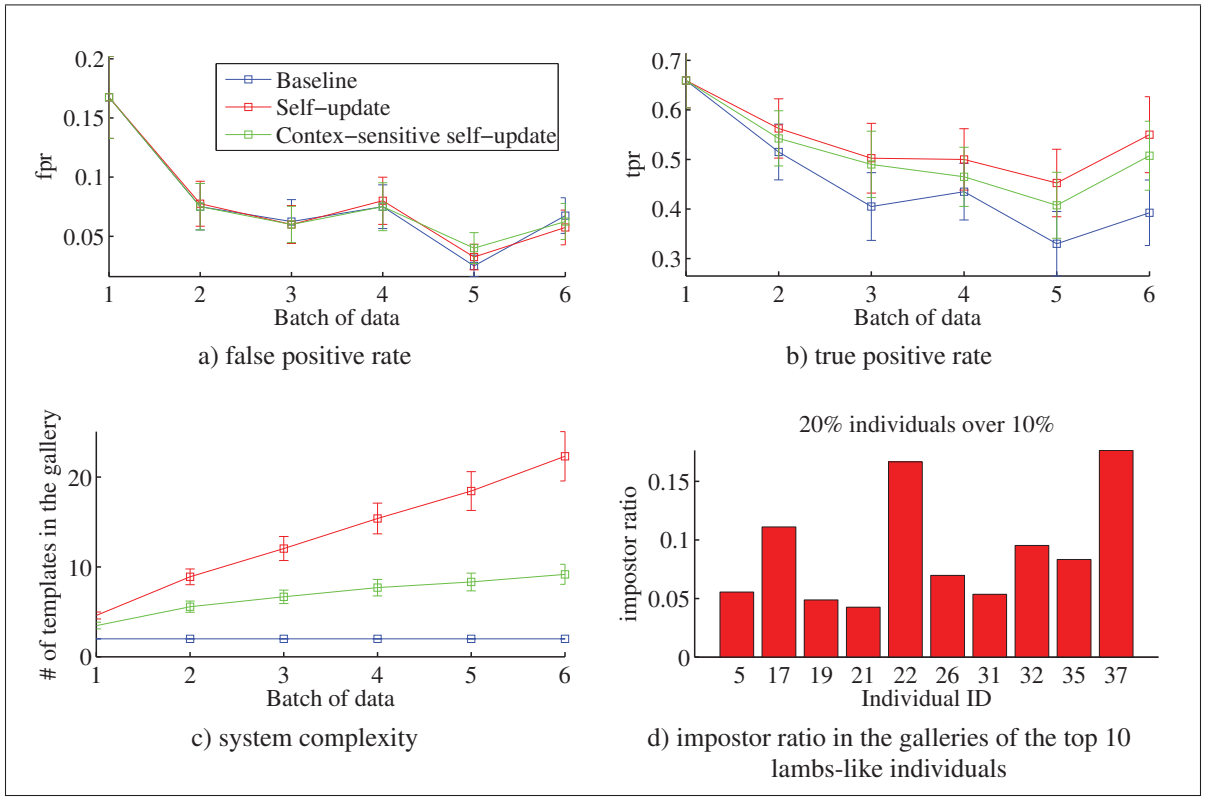


Figure 2.8 Simulation results with DIEE dataset where the updating threshold is selected for fpr $= 1\%$.

Figure 2.8 presents the average performance results for the fpr $= 1\%$ updating thresholds for the self-updating techniques. An overall performance increase is shown for the self-updating methods. A higher tpr is observed throughout the entire simulation, ending at tpr $= 55.0 \pm 7.7\%$ for self-updating, and tpr $= 50.8 \pm 7.0\pm$ for *context-sensitive* self-updating (see Fig. 2.8 (b)).

While results with self-updating are higher in this application, it is important to note that improvements come at the expense of a doubled average gallery size (see Fig. 2.8 (c)), as well as an increase in the impostor ratio (see Fig. 2.8 (d), 20% of the galleries are composed by more than 10% impostor templates). Comparing these ratios with the the previous ones (in Fig. 2.7), it is apparent that this increase is not connected to specific lamb-type individuals, but to all the enrolled individuals. This underlines the importance of updating thresholds, specifically for long-term operations where the impostor ratio would be likely to grow exponentially as the facial models become corrupted.

2.5.2 Video Surveillance with FIA Data

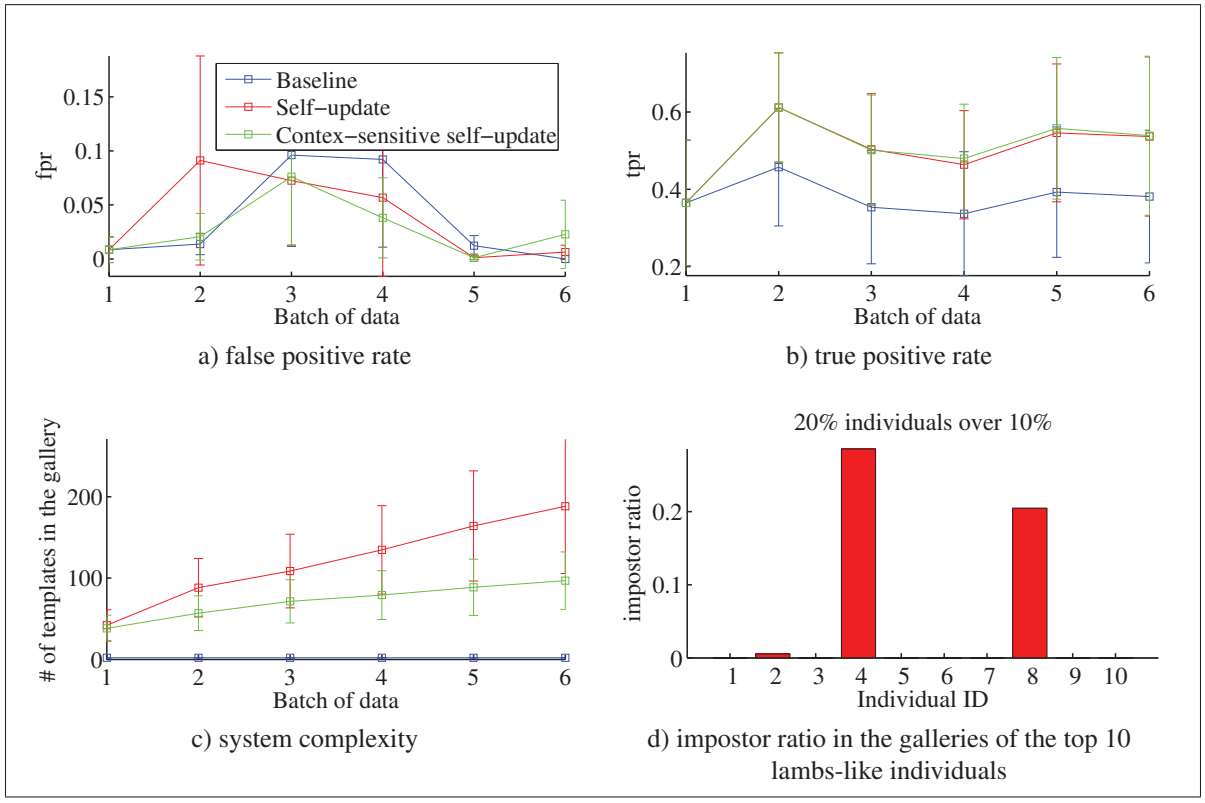


Figure 2.9 Simulation results with FIA dataset where the updating threshold is selected for fpr = 0%.

Figure 2.9 presents the average performance results for the fpr $= 0\%$ updating thresholds for the self-updating techniques. In this scenario involving more sources of variations in capture conditions than the DIEE dataset (see Table 2.1), the benefits of a self-updating strategy are more significant, as the self-updating systems exhibit a significantly higher tpr during the entire simulation (see Fig. 2.9 (b)). From batch 2 to 6, the self-updating systems are stable close to tpr $= 60\%$ (both ending at $53 \pm 20\%$), while the baseline system remains close to tpr $= 40\%$ (ending at $38.1 \pm 17.2\%$).

As a consequence of the more complex nature of a semi-controlled surveillance environment as well as the higher number of facial captures, performance improvements come at the expanse of significantly larger galleries than with the DIEE dataset (see Fig. 2.9 (c)), ending at an average of 188 ± 83 templates for self-update, and 97 ± 35 templates for *context-sensitive* self-update. It can still be noted that the filtering strategy of the *context-sensitive* self-update technique enables to maintain a comparable level of performance, for gallery sizes approximately two times smaller.

Among the 10 individuals of interest, 2 lamb-like individuals (ID 4 and 8) can be identified, with an impostor ratio over 20% (see Fig. 2.9 (d)). Despite the added complexity of a semi-constrained environment, the higher number of faces captured in video streams enables a better definition of facial models of target individuals during the first batch. This explains that impostor templates have only be added to two difficult lamb-type individuals, and not all the galleries.

In Figure 2.10 (b), it can be observed that a more relaxed fpr $= 1\%$ constraint for the updating threshold didn't have a significant impact on the performance of self-updating systems. However, the average gallery size of the self-updating technique increased to end at 268 ± 71 templates, while the *context-sensitive* self-updating technique enabled to remain at a lower size of 109 ± 38 templates (see Fig. 2.10 (c)), comparable to the fpr $= 0\%$ threshold results (see Fig. 2.9 (c)). This observation reveals that a majority of the new templates added with the fpr $= 1\%$ thresholds contained redundant information, that was already present in the galleries.

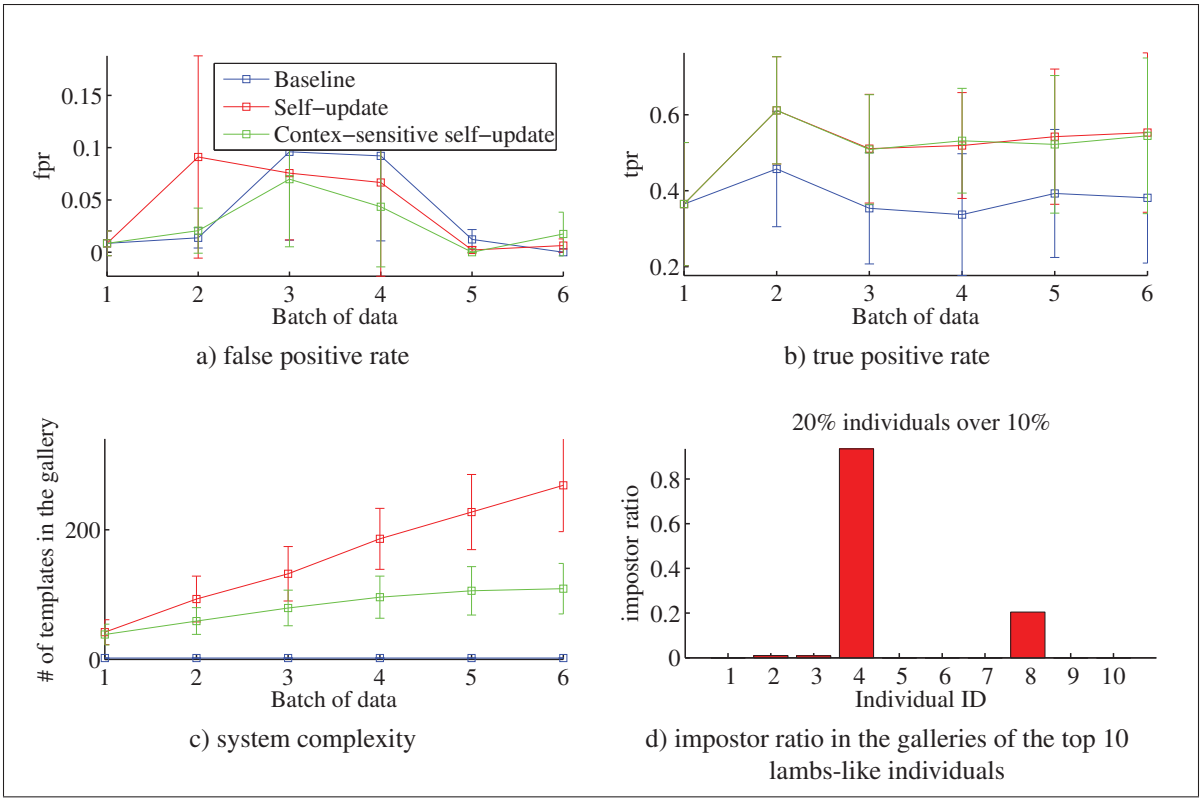


Figure 2.10 Simulation results with FIA dataset where the updating threshold is selected for fpr = 1%.

This underscores the benefits of the *context-sensitive* self-updating technique when operating with videos, where higher quantities of templates may be selected for self-updating. By reducing the number of updates, this technique enables to mitigate the growth in computational complexity of the prediction process as well as the need to use a costly template-management system, without impacting system performance.

Impostor ratios in Fig. 2.10 (d) show a significant increase for individual ID 8, which ends at 80%. This confirms the rapid addition of impostor templates to the galleries in long term operations. In this video-surveillance scenario where more facial captures are presented to the system (compared to the DIEEE scenario), the gallery of lamb-like individual 4 is updated with a larger amount of impostor templates at the beginning of the simulation. This gallery then keeps attracting impostor templates over time, which reduces the pertinence of the facial model.

2.5.3 Unconstrained Face Recognition with FRGC Data

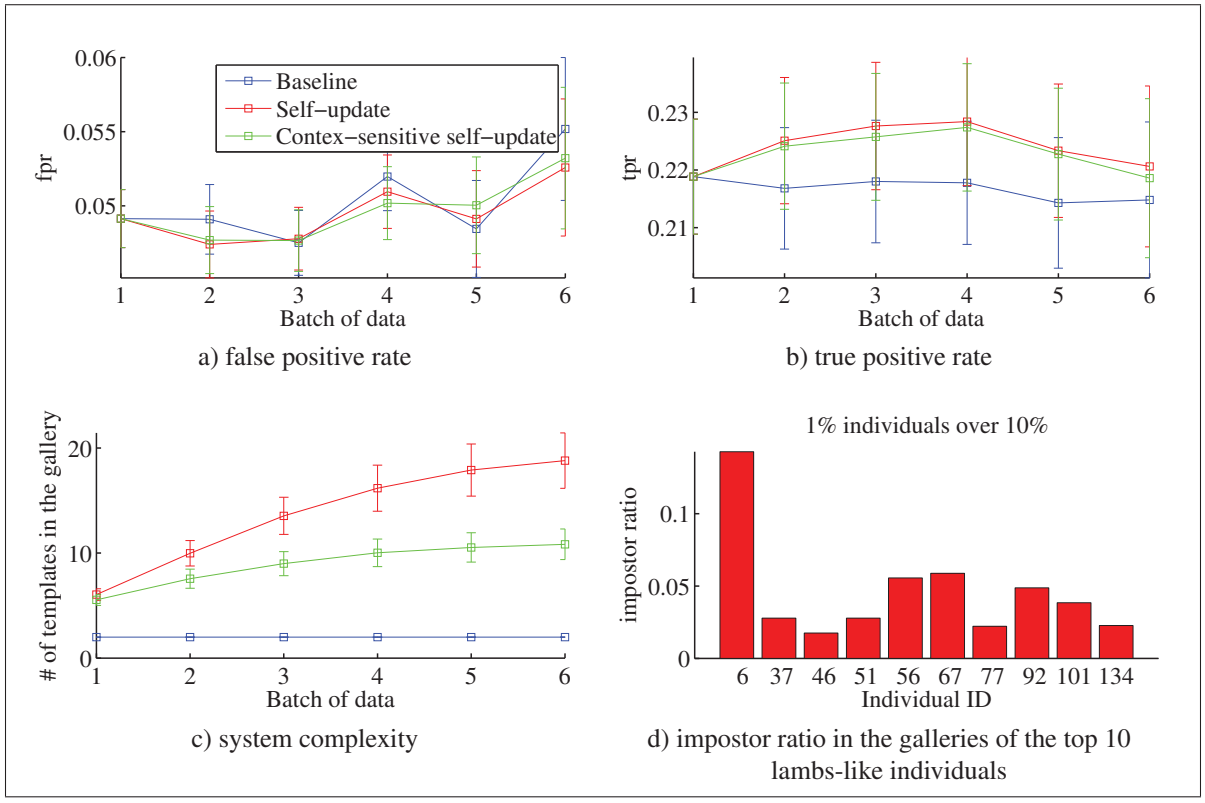


Figure 2.11 Simulation results with FRGC dataset where the updating threshold is selected for fpr $= 0\%$.

Figure 2.11 presents the average performance results for the fpr $= 0\%$ updating thresholds for the self-updating techniques. It can be observed in Fig. 2.11 (b) that this scenario represents a significantly harder FR problem, as all three systems perform below tpr $= 23\%$ during the entire simulation. In addition, despite the increase in average gallery size up to respectively 18.8 ± 2.7 and 10.8 ± 1.5 templates for the self-update and *context-sensitive* self-update techniques (see Fig. 2.11 (c)), only a marginal performance gain can be observed. The two self-updating systems end at tpr $= 22.1 \pm 1.4\%$ and tpr $= 21.9 \pm 1.4\%$, while the baseline case exhibits a tpr $= 21.5 \pm 1.4\%$.

A bigger impact can be observed in Fig. 2.12 (b), presenting tpr performance of the three systems for the fpr $= 1\%$ updating threshold. From batch 2 to 6, the two self-updating cases

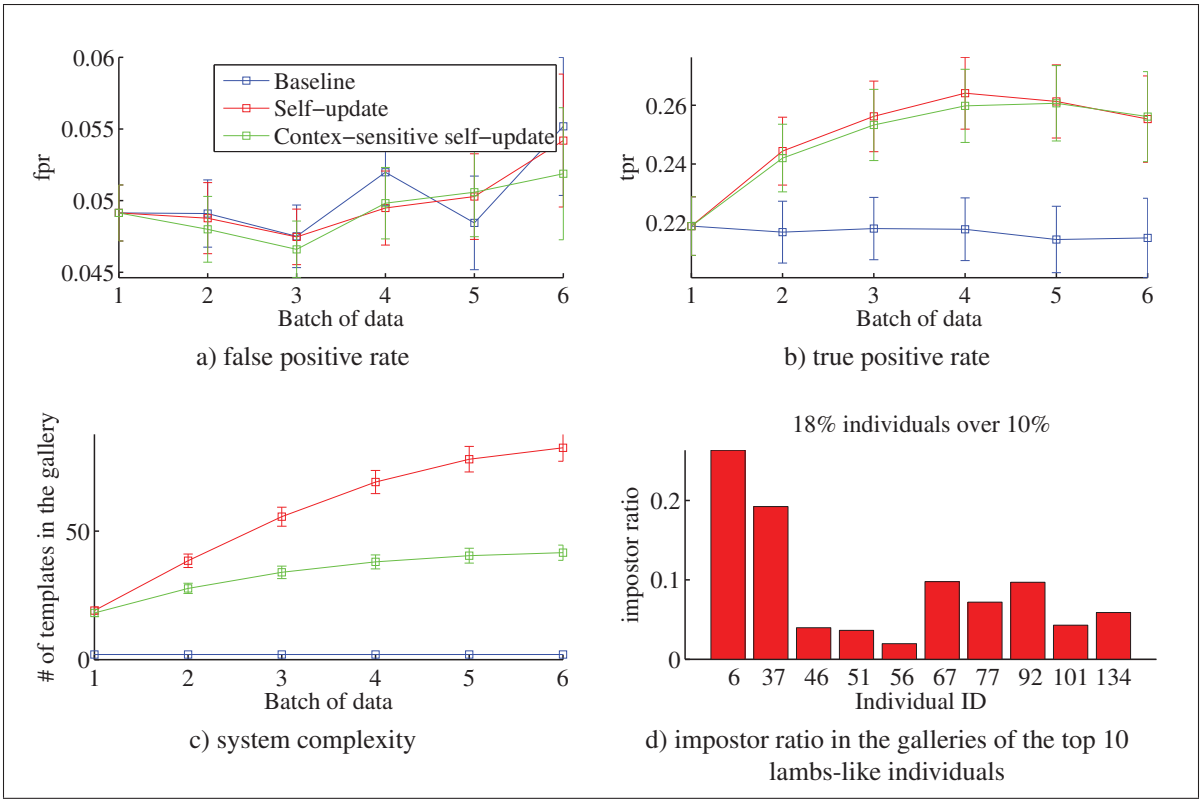


Figure 2.12 Simulation results with FRGC dataset where the updating threshold is selected for fpr $= 1\%$.

present significantly higher tpr performance, both ending at tpr $= 25.6 \pm 1.5\%$. However, as in the previous scenarios, this performance gain comes at the expense of a significantly higher system complexity. Both systems with self-update end with respectively 82.4 ± 5.2 and 41.6 ± 2.0 templates in the galleries (see Fig. 2.12 (c)). The average impostor ratio also increased significantly, as 18% of the galleries contain more than 10% impostor templates (see Fig. 2.12 (d)), while only 1% of the galleries were in this situation with the fpr $= 0\%$ updating threshold.

Results are related to the nature of the scenario presented in Section 2.4.1.3. The multiple enrolment sessions (up to 16), where small numbers of ROI were captured (6 ROIs), favour the presence of genuine captures that are different enough to fail the updating threshold test. Fewer than 20 templates per individuals have been added to self-updating galleries with $far = 0\%$ self-updating threshold (see Fig. 2.11 (c)), despite the presence of more than 100 genuine samples in batches. In addition, the systems are initialized with the first capture session, where

only 4 controlled stills are available to build the facial model before processing uncontrolled captures in future sessions. This prevents the generation of representative facial models, that either reject a majority of genuine templates, or accept a significant amount of impostor templates depending on the updating threshold (see Fig. 2.12 (d)).

Despite the improved performance achieved using self-updating techniques, this dataset raises the limitations of using a self-updating system relying on a two-threshold update strategy in complex environments, with limited reference data and uncontrolled variations in capture conditions.

2.5.4 Summary and Discussions

In all experimental results, the following general observations have emerged:

a. Both self-updating techniques generate a significant and stable performance boost over time.

b. The template filtering strategy of the proposed *context-sensitive* self-updating technique significantly reduces system complexity. The galleries are approximately 2 times smaller than a standard self-updating system, without impacting performance.

c. Using a less stringent constraint of $\text{fpr} = 1\%$ for the updating threshold does not always have an impact on the performance boost, but always increases system complexity as well as the number of impostor templates in the galleries.

While these observations remain valid for each scenario, a more precise analysis reveals potential limitations of these approaches depending on the represented application.

In a semi-controlled FR application with limited changes mainly caused by illumination and expression (DIEE dataset), benefits of a self-updating techniques are quite clear. In fact, despite the increase in the number of impostor samples in the gallery, a significant performance boost can be observed when a more relaxed updating-threshold is selected.

In the case of a video-surveillance scenario involving a higher amount of impostor individuals not modelled by the system (FIA dataset), a more relaxed updating threshold didn't show any performance improvement, despite a doubled average gallery size for the self-updating technique (while the *context-sensitive* self-update technique prevented any increase in average gallery size). While the overall performance wasn't lowered, the gallery of one specific individual was severely affected, ending with around 80% of impostor samples. In such scenario, involving multiple causes of variation (face angle, resolution, motion blur, etc.) as well as a greater amount of impostor individuals, manual intervention may be necessary at regular intervals, to ensure that the gallery of some specific individuals (lambs) are not getting corrupted over time.

Finally, in the more complex scenario represented by the FRGC dataset, the performance gain observed with the self-updating techniques was considerably lower, even with the less stringent updating threshold of $far = 1\%$. In this scenario, systems are presented with significantly different samples in early operations (after the 4th image), as opposed to the DIEE and FIA scenarios (with respectively 10 and around 30 samples for a first session). In such application, a manual intervention may be required at the early stages of operations, to ensure that the facial models are initialized with enough representative templates to be able to keep updating over time.

2.6 Conclusion

Despite the advances in feature extraction and classification techniques, face recognition in changing environments remains a challenging pattern recognition problem. Changes in capture condition or individuals physiology can have a significant impact on a system performance, where initial facial models are often designed with a limited amount of reference templates, and frequent re-enrolment sessions are not always possible. Adaptive classification techniques have been proposed in the past decade to address this challenge, relying on operational data to adapt the system over time. Among them, self-updating techniques have been proposed for automatic adaptation using highly-confident captures labelled by the system. While this enables to

automatically benefit from a considerable source of new information without requiring a costly manual updating process, these systems are particularity sensitive to their internal parameters. A trade-off between assimilation of new information and protection against the corruption of facial models with impostor templates has to be considered, as well as a limitation of system complexity over time. While template management techniques can be used to limit system complexity, they remain costly and may interfere with seamless operations.

In this chapter, self-updating methods have been surveyed in the context of a face recognition application with template matching. A *context-sensitive* self-update technique has been presented to limit the growth in system complexity over time, by relying on additional information related to the capture conditions. With this technique, only highly-confident faces captured under new conditions are selected to update individual facial models, effectively filtering out redundant information. A specific implementation of a template matching system with *context-sensitive* self-update has been proposed, where changes are detected in illumination conditions. Proof-of concept experimental simulations using thee publicly-available face databases showed that this technique enables to maintain the same level of performance than a regular self-updating template matching system, with a significant gain in terms of memory complexity. By using additional information available in the face captures during operations, this technique allows to reduce the size of template galleries by half, effectively mitigating the computational complexity of the recognition process over time. In applications where memory footprint has to be restricted, this strategy would also limit the need to use costly template management techniques during operations.

However, application-specific limitations have been observed during simulations. When faced with recognition environments with significant variations, and a limited pool of reference patterns for initial enrolment, self-updating systems can be very sensitive to the initialization of their template galleries, as well as the updating threshold. A stricter updating rule may be required to prevent updating with impostor samples, which can significantly reduce the benefits of a self-updating strategy that would never detect any highly confident samples. In addition, while the proposed *context-sensitive* self-updating techniques enabled to significantly reduce

system complexity, it relies on the storage of input ROIs in addition to reference patterns in the galleries, as well as an additional measurement during operations.

While self-updating techniques can significantly improve the recognition performance of face recognition systems, their implementation should always be tailored to the specificities of the application as well as the recognition environment. While human intervention can be reduced with automatic strategies, it will still plays a significant role in certain applications, especially when dealing with significant variations in capture conditions. In those cases, occasional manual confirmation should be considered, in order to maintain the system's performance by adapting to abrupt changes.

CHAPTER 3

ADAPTIVE ENSEMBLES FOR FACE RECOGNITION IN CHANGING VIDEO SURVEILLANCE ENVIRONMENTS

Christophe Pagano[1], Eric Granger[1], Robert Sabourin[1], Gian Luca Marcialis[2], Fabio Roli[2]

[1] Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle, École de Technologie Supérieure, Université du Québec, Montréal, Canada

[2] Patter Recognition and Applications Group, Department of Electrical and Electronic Engineering, University of Cagliary, Italy

Article published in « Information Sciences » by Elsevier, in 2014.

Abstract

Recognizing faces corresponding to target individuals remains a challenging problem in video surveillance. Face recognition (FR) systems are exposed to videos captured under various operating conditions, and, since data distributions change over time, face captures diverge w.r.t. stored facial models. Although these models may be adapted when new reference videos become available, incremental learning with faces captured under different conditions may lead to knowledge corruption. This paper presents an adaptive multi-classifier system (AMCS) for video-to-video FR in changing surveillance environments. During enrolment, faces captured in reference videos are employed to design an individual-specific classifier. During operations, a tracker allows to regroup facial captures for individuals in the scene, and accumulate the predictions per track for robust spatiotemporal FR. Given a new reference video, the corresponding facial model is adapted according to the type of concept change. If a gradual pattern of change is detected, the individual-specific classifier(s) are adapted through incremental learning. To preserve knowledge, another classifier is learned and combined with the individual's previously-trained classifier(s) if an abrupt change is detected. For proof-of-concept, the performance of a particular implementation of this AMCS is assessed using videos from the Faces in Action dataset. By adapting facial models according to changes detected in new reference

videos, this AMCS allows to sustain a high level of accuracy comparable to the same system that is always updated using a learn-and-combine approach, while reducing time and memory complexity. It also provides higher accuracy than incremental learning classifiers that suffer the effects of knowledge corruption.

3.1 Introduction

The global market for video surveillance (VS) technologies has reached revenues in the billions of $US as traditional analogue technologies are replaced by IP-based digital ones. VS networks are comprised of a growing number of cameras, and transmit or archive massive quantities of data for reliable decision support. The ability to automatically recognize and track individuals of interest across these networks, and under a wide variety of operating conditions, may provide enhanced screening and situation analysis.

In decision support systems for VS, face recognition (FR) has become an important function in two types of applications. In *watch-list screening applications*, facial models[1] used for classification are designed using regions of interest (ROIs) extracted from the reference still images or mugshots of a watch-list. Then, still-to-video FR seeks to determine if faces captured in video feeds correspond to an individual of interest. In *person re-identification* for search and retrieval applications, facial models are designed using ROIs extracted from reference videos and tagged by a human operator. Then, video-to-video FR seeks to alert the operator when these individuals appear in either live (real-time monitoring) or archived (post-event analysis) videos.

This paper focuses on the design of robust classification systems for video-to-video FR in changing surveillance environments, as required in person re-identification. In this context, public security organizations have deployed many CCTV and IP surveillance cameras in recent years, but FR performance is limited by human recognition abilities. Indeed, accurate and

[1] A *facial model* is defined as either a set of one or more reference captures (used in template matching systems), or a statistical model estimated through training with reference captures (used in neural or statistical classification systems) corresponding to a target individual.

timely recognition of ROIs is challenging under semi-controlled (e.g., in an inspection lane, portal or checkpoint entry) and uncontrolled (e.g., in cluttered free-flow scene at an airport or casino) capture conditions. Given the limited control during capture, the performance of state-of-the-art systems are affected by the variations of pose, scale, orientation, expression, illumination, blur, occlusion and ageing. Moreover, FRiVS is an open set problem, where only a small proportion of the faces captured during operations correspond to individuals of interest. Finally, ROIs captured in videos are matched against facial models designed a priori, using a limited number of high quality reference samples captured during enrolment. Accuracy of face classification is highly dependent on the representativeness of models, and thus number, relevance and diversity of these samples.

Some specialized classification architectures have been proposed for FRiVS. For instance, the open-set Transduction Confidence Machine-*k*NN (TCM-kNN) is comprised of a global multi-class classifier with a rejection option tailored for unknown individuals (Li and Wechsler, 2005). Classification systems for FRiVS should however be modeled as independent individual-specific detection problems, each one implemented using one- or two-class classifiers (i.e., detectors), with specialized thresholds applied to their output scores (Pagano *et al.*, 2012). The advantages of class-modular architectures in FRiVS (and biometrics in general) include the ease with which facial models (or classes) may be added, updated and removed from the systems, and the possibility of specializing feature subsets and decision thresholds to each specific individual. Individual-specific detectors have been shown to outperform global classifiers in applications where the reference design data is limited w.r.t. the complexity of underlying class distributions and to the number of features and classes (Oh and Suen, 2002; Tax and Duin, 2008). Moreover, some authors have argued that biometric recognition is in essence a multi-classifier problem, and that biometric systems should co-jointly solve several classification tasks in order to achieve state-of-the-art performance (Bengio and Mariethoz, 2007).

Modular architectures for FRiVS have been proposed by Ekenel et al. (Ekenel *et al.*, 2009), where 2-class individual-specific Support Vector Machines are trained on a mixture of target

and non-target samples. Given the limited amount of reference samples and the complexity of environments, modular approaches have been extended by assigning a classifier ensemble to each individual. For example, Pagano et al. (Pagano *et al.*, 2012) proposed a system comprised of an ensemble of 2-class ARTMAP classifiers per individual, each one designed using target and non-target samples. A pool of diversified classifiers is generated using an incremental learning strategy based on Dynamic PSO, and combined in the ROC space using a Boolean fusion function.

In person re-identification, new reference video become available during operations or through some re-enrolment process, and an operator can extract a set of facial ROIs belonging to a target individual. In order to adapt an individual's facial model in response to these new ROI samples, the parameters of a individual-specific classifier can be re-estimated through supervised incremental learning. For example, ARTMAP neural networks (Carpenter *et al.*, 1992) and extended Support Vector Machines (Ruping, 2001) have been designed or modified to perform incremental learning. However, these classifiers are typically designed under the assumption that data is sampled from a static environment, where class distributions remain unchanged over time (Granger *et al.*, 2008).

Under semi- and uncontrolled capture conditions, ROI samples that are extracted from new reference videos may incorporate various patterns of change that reflect varying concepts[2]. While gradual patterns of change in operational conditions are often observed (due to, e.g., ageing over sessions), abrupt and recurring patterns (caused by, e.g., new pose angle versus camera) also occur in FRiVS. A key issue in changing VS environments is adapting facial models to assimilate samples from new concepts without corrupting previously-learned knowledge, which raises the *plasticity-stability* dilemma (Grossberg, 1988). Although updating a single classifier may translate to low system complexity, incremental learning of ROI samples extracted from videos that reflect significantly different concepts can corrupt the previously acquired knowledge (Connolly *et al.*, 2012; Polikar *et al.*, 2001). Incomplete design data and changing

[2] A *concept* can be defined as the underlying class distribution of data captured under s specific condition, in our context due to different pose angle, illumination, scale, etc. (Narasimhamurthy and Kuncheva, 2007).

distributions contribute to a growing divergence between the facial model and the underlying class distribution of an individual.

Adaptive ensemble methods allow to exploit multiple and diverse views of an environment, and have been successfully applied in cases where concepts change in time. By assigning an adaptive ensemble to each individual, it is possible to adapt a facial model by updating the pool of classifiers and/or the fusion function (Kuncheva, 2004b). For example, with iques like Learn++ (Polikar *et al.*, 2001) and other Boosting variants (Oza, 2001), a classifier is trained independently using new samples, and weighted such that accuracy is maximized. Other approaches discard classifiers when they become inaccurate or concept change is detected, while maintaining a pool with these classifiers allows to handle recurrent change. Classifier ensembles are well suited for adaptation in changing environments since they can manage the *plasticity-stability* dilemma at the classifier level – when samples are significantly different, previously acquired knowledge can be preserved by initiating and training a new classifier on the new data. However, since the number of classifiers grows, benefits (accuracy and robustness) are achieved at the expense of system complexity.

In this paper, an adaptive multi-classifier system (AMCS) for video-to-video FR is proposed to maintain a high level of performance in changing surveillance environments. It is initially comprised of a single two-class incremental learning classifier (or detector) per individual, and a change detection mechanism. During enrolment of an individual, ROI samples are extracted from a reference video sequence, and employed to initiate and train the detector. Then, during operations, a face tracker is used to regroup ROIs for different people in the scene according to trajectory[3]. For robust spatio-temporal FR, the prediction of each individual-specific detector is accumulated over along different trajectories. The proposed system allows to update the facial model (detectors) of an individual in response to a new reference video. The change detection mechanisms determines the extent to which ROI samples of a trajectory extracted from new videos correspond to previously-learned concepts. To limit system complexity, if

[3] A *facial trajectory* is defined as a set of ROIs (isolated through face detection) that correspond to a same high quality track of an individual across consecutive frames.

ROI samples incorporate a gradual pattern of change w.r.t. existing concepts, the corresponding pool of classifiers (and, if needed, fusion function) are updated through incremental learning. In contrast, to avoid issues related to knowledge corruption, the AMCS employs a learn-and-combine approach if ROI samples exhibit an abrupt pattern of change w.r.t. existing concepts. Another dedicated classifier is initiated and trained on the new data, and then combined with the individual's previously-trained classifiers.

Some approaches in literatures also exploit change detection to drive adaptation or online-learning of classification systems, such as the Diversity for Dealing with Drifts algorithm (Minku and Yao, 2012), the incremental learning strategies based on dynamic PSO (Connolly *et al.*, 2012), and a Just-in-Time architecture that regroups reference templates per concept (Alippi *et al.*, 2013). However these approaches adapt to changing environments by focusing on the more recent concepts, though weighing or by discarding of previously-learned concepts. In the proposed system, change detection allows to compromise between *stability* (adapting classifiers to known concepts) and *plasticity* (generation of classifiers for new concepts), thereby preserving knowledge (and the ability to recognize) for previously-learned and recurring concepts.

For validation, a particular implementation of the AMCS was considered. During the enrolment of an individual, an histogram representation of the ROI sample distribution is stored, and an incremental learning strategy based on Dynamic Particle Swarm Optimization (DPSO) (Connolly *et al.*, 2012) is employed to generate and evolve a diversified pool of 2-class ART-MAP classifiers (Carpenter *et al.*, 1992) using a mixture of target (individual) and non-target (universal and cohort model) samples. Then, when a new reference video (trajectory) becomes available, the change detection process evaluates whether its ROI samples exhibit gradual or abrupt changes w.r.t. to all previously stored histogram distributions using the Hellinger Drift Detection Method (Ditzler and Polikar, 2011). If the new reference samples exhibit a gradual change, the classifier trained with similar data is updated and re-optimized through the DPSO-based learning strategy. If the new reference samples present a significant change, a new histogram distribution is stored, and a new pool of classifiers is generated and optimized.

For each pool, the best classifier is then selected to represent its corresponding concept in the AMCS. Each target individual is associated with a single classifier or an ensemble of classifiers, where outputs are combined using a weighted-average score fusion rule. The accuracy and resource requirements of this system is assessed using facial trajectories extracted from video surveillance streams of the Face in Action database (Goh *et al.*, 2005). It us comprised of over 200 individuals captured over several months, exhibiting gradual (e.g. expression, ageing) and abrupt (e.g. orientation, illumination) changes.

The rest of this paper is structured as follows. The next section briefly reviews the techniques and challenges of FRiVS. Then, an overview of the literature on change detection and adaptive biometrics is presented in Section 3.3. In Section 4, the adaptive MCS proposed for video-to-video FR is presented. The experimental methodology (video data, protocol and performance measures) used for validation is presented in Section 5. Finally, simulation results are presented and discussed in Section 6.

3.2 Background – Face Recognition in Video Surveillance

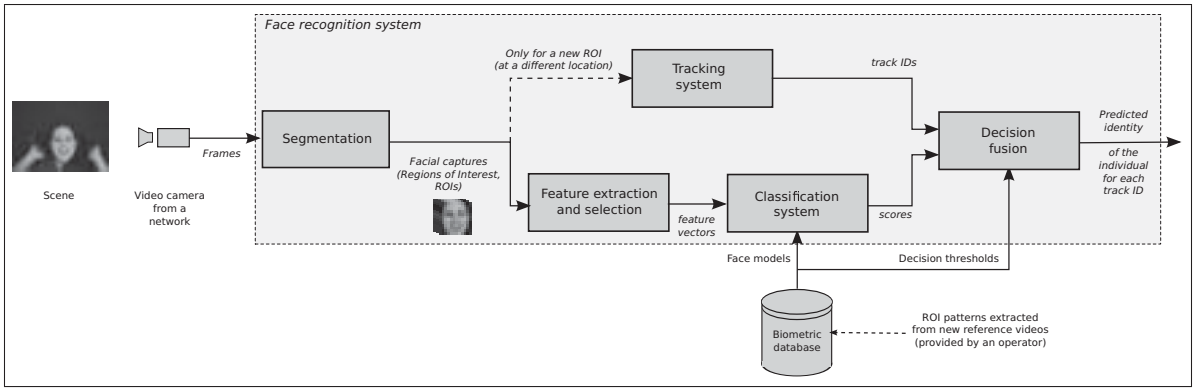


Figure 3.1 A human centric system face video-to-video face recognition.

The problem addressed in this paper is the design of accurate and robust systems for video-to-video FR under semi- and uncontrolled capture conditions. Assume that FR is embedded as software executing inside some human-centric decision support system for intelligent video

surveillance. In person re-identification applications, an operator may enroll an individual of interest appearing in a video sequence, and gradually design and update their facial models over time by analyzing one or more reference video feeds captured from the particular scene or other sources. Then, individuals of interest must be detected over a network of digital surveillance cameras by matching facial captures against their facial models.

3.2.1 A generic system for video face recognition

Figure 3.1 presents a generic system for video-to-video FR. Each camera captures streams of 2D images or frames, and provides the system with a particular view of individuals populating the scene. This system first performs segmentation to isolate ROIs corresponding to the faces in a frame, from which invariant and discriminant features are extracted and selected for classification (matching) and tracking functions. Some features are assembled into an input pattern, $\mathbf{q} = (q_1, ..., q_D)$ for classification, and pattern $\mathbf{b} = (b_1, ..., b_e)$ for tracking.

During enrolment, a sot of one or more reference patterns $\mathbf{a}^i[t]$ corresponding to an individual i are extracted from captured ROIs on one or more reference video streams provided by the operator at time t. These are employed to design the user's specific facial model to be stored in a biometric database, as a template or a statistical model. Recognition is typically implemented using a template matcher or using a neural or statistical classifier trained a priori to map the reference patterns to one of the predefined classes, each one corresponding to an individual enrolled to the system. Although each facial model may consist of a set of one or more templates (reference ROI patterns) for template matching, this paper assumes that a model consists of parameters estimated by training a classifier on the reference ROI patterns.

During operations, ROI patterns extracted for unknown individuals in the scene are matched against the model of individuals enrolled to the system. The resulting classification score $s_i(\mathbf{q})$ indicates the likelihood that pattern $\mathbf{q}$ corresponds to the individual i, for $i = 1, ..., I$. Each score is compared against the user-specific decision thresholds, θ^i, and the system outputs a list of all possible matching identities. To reduce ambiguities during the decision process, the face

tracker may follow the motion and appearance of faces in the scene over successive frames. This allows to regroup ROIs of different individuals and accumulate their matching scores over time.

3.2.2 State-of-the-art in video surveillance

A common approach to recognizing faces in video consists in only exploiting spatial information, and applying techniques for still-to-still FR (like Eigenfaces or Elastic Bunch Graph Matching) only on high quality ROIs isolated during segmentation (Zhao *et al.*, 2003). FRiVS remains a difficult task since the faces captured in video frames are typically lower quality and generally smaller than still images. Furthermore, faces captured from individuals in a semi- or unconstrained environment may vary considerably due to limited control over capture conditions (e.g., illumination, pose, expression, resolution and occlusion), and due to changes in an individual's physiology (e.g., aging) (Matta and Dugelay, 2009). Given these difficulties, high quality faces may never be captured or recognized. More powerful front end processing (face capture and representation) and back-end processing (fusion or responses from cameras, templates, frames) is required for robust performance.

Despite these challenges of video-based FR, it is possible to exploit spatio-temporal information extracted from video sequences to improve performance (see Fig. 3.1). As mentioned, using face tracking, evidence in individual frames can be integrated over video streams, potentially leading to improved robustness and accuracy. For example, track-and-classify approaches combine information from the motion and appearance faces in a scene to reduce ambiguity (e.g., partial occlusion) (Barry and Granger, 2007).

Beyond spatio-temporal approaches, specialized classification architectures have also been proposed for accurate FRiVS. In this case, *open-set* or *open-world* FR operates under the assumption that most faces captured during operations do not correspond to an individual of interest (Li and Wechsler, 2005). The probability of detecting the presence of a restrained group of individuals of interest in scenes may be quite low, and facial models may incorporate a signif-

icant amount of uncertainty w.r.t. operational environments (Committee *et al.*, 2010; Rattani, 2010).

The Transduction Confidence Machine *k*-Nearest Neighbour (TCM-*k*-NN) has been proposed for open-set FR using a multi-class architecture and a specialized rejection option for individuals not enrolled to the system (Li and Wechsler, 2005). Kamgar-Parsi et al. propose a face space projection technique where a feed-forward network is designed for each individual (Kamgar-Parsi *et al.*, 2011). In addition, some multi-verification architectures based on with an individual-specific reject option have been proposed by Stallkamp et al. (Ekenel *et al.*, 2009) and by Tax and Duin (Tax and Duin, 2008), where a specific one- or two-class classifier is assigned to each individual enrolled to the system. These systems allow to add and remove an individual without requiring a complete re-design of the system, and to select individual specific thresholds and feature sets (Tax and Duin, 2008). This ability is particularly favourable in person re-identification, where new individuals are enrolled and monitored on-the-fly by the operator. In addition, separating a multi-class classification problem into more treatable one or two-class problems has been shown to improve the overall performance of the system, adopting the "divide and conquer" approach. For example, in a comparison of classification architectures for FRiVS based on ARTMAP neural networks, class-modular architectures exhibited significant performance improvements (Pagano *et al.*, 2012). Similarly, in other biometric applications such as character recognition, the performance of a Multi-Layer Perceptron have been significantly improved by the introduction of a class-modular architecture (Oh and Suen, 2002) (Kapp *et al.*, 2007).

Finally, several other biometric applications, such as speech recognition, operate in *open-set* environments, and exploit a universal background model (UBM) – a non-target population to generate samples from unknown persons from which the target individual can be discriminated – as well as cohort models (CMs) – a non-target population of other people enrolled in the system (Brew and Cunningham, 2009). The use of such CM is of interest in class-modular architectures such as (Tax and Duin, 2008; Ekenel *et al.*, 2009). Sharing information among the different target persons in a class-modular architecture is necessary in order to achieve a high

level of performance (Bengio and Mariethoz, 2007). Indeed, using some common reference samples (target and non-target samples from a same CM) to design classifiers can be considered as information sharing between classifiers, and may improve the overall system performance.

3.2.3 Challenges

Systems for FRiVS encounter several challenges in practice. In particular, the facial models are often poor representatives of faces to be recognized during operations (Rattani, 2010). They are typically designed during an a priori enrolment phase, using limited number of reference ROI patterns $\mathbf{a}^i[t]$ from new sets of samples, linked to unknown probability distributions $p(\mathbf{a}[t]|i)$. The underlying data distribution corresponding to individuals enrolled to the system is complex mainly due to: (1) inter- and intra-class variability, (2) variations in capture conditions (interactions between individual and camera), (3) the large number of input features and individuals, and (4) limitations of cameras and signal processing techniques used for segmentation, scaling, filtering, feature extraction and selection, and classification (Committee *et al.*, 2010). The performance of FR systems may decline considerably because state-of-the-art neural and statistical classifiers depend heavily on the availability of representative reference data for design and update of facial models. In addition, the probability distribution may change gradually or abruptly over time. All these factors contribute to a growing divergence between the facial model of an individual and its underlying class distribution.

Although limited reference data is initially available to design facial models, new reference video sequences may become available over time in a person re-identification application. The systems proposed in the literature for FRiVS usually focus on the matching accuracy, facial quality and the *open-set* context, but not on the update of the facial models with ROIs from new and diverse reference videos.

In semi- or uncontrolled environments, faces captured for an individual can correspond to several *concepts* in the input feature space, which can all be relevant for different capture conditions during system operation. Reference video sequences may incorporate samples cor-

responding to different capture conditions, such as pose angles, illumination, and even ageing. While updating the facial models with new videos from known concepts can reinforce the system's knowledge, incremental learning of new reference videos that incorporate different concepts can be a challenge. For example, updating a system with ROI patterns with a specific pose angle can corrupt previously-learned knowledge, learned from samples with other angles. A robust system for FRiVS should detect the presence of various types of changes in the underlying data distribution of individuals. When a new concept emerges, a suitable update strategy should be triggered to preserve pre-existing concepts.

3.3 Concept Change and Face Recognition

In this paper, a mechanism is considered to detect changes in the underlying data distribution from new reference videos. This mechanism will then trigger different updating strategies. Concept change has been defined by several authors in statistical pattern recognition literature (Kuncheva, 2004a). A *concept* can be defined as the underlying data distribution in $\mathbb{R}^D$ of the problem at some point in time (Narasimhamurthy and Kuncheva, 2007). Given a set of reference ROIs $\{\mathbf{a}^i[t]\}$ captured from target individual i at time t (sampled from the underlying class distribution), a class-conditional distribution of data $p(\mathbf{a}[t]|i)$ may be defined. A *concept change* encompasses various types of noise, trends and substitutions in the underlying data distribution associated with a class or concept. The main assumption is the uncertainty about the future: the data distribution from which the future instance is sampled, $p(\mathbf{a}_{t+1}|i)$ is unknown. To simplify the notation, the time t will be omitted for the remaining of this section, but all the data distribution will be assumed to be time dependent.

A statistical pattern recognition problem can incorporate change due to class priors, $p(i)$, class-conditional distributions $p(\mathbf{a}|i)$ and posterior distributions $p(i|\mathbf{a})$ (Kuncheva, 2004a). A categorization of changes has been proposed by Minku et al. (Minku *et al.*, 2010), based on severity, speed, predictability and number of re-occurrences, but the following four categories are mainly considered in the literature: noise and abrupt, gradual and recurring changes (Kuncheva, 2008).

Table 3.1 Types of changes occurring in FRiVS environments.

Type of change	Examples in face recognition
Static environment with: – random noise – hidden contexts	 – inherent noise of system (camera, matcher, etc.) – different known view points from a camera or of a face (e.g. illumination of images, new face pose or orientation) (Figure 3.2 (a))
Dynamic environment with: – gradual changes – sudden abrupt changes – recurring contexts	 – aging of user (Figure 3.2 (b)) –new unknown view points on traits; change of camera (Figure 3.2 (a)) – unpredictable but recurring changes in capture conditions (e.g. lighting changes due to the weather) (Figure 3.2 (c))

Concept changes in pattern recognition may be viewed in the context of FRiVS, where changes can originate from variations in an individual's physiology, as well as in observation conditions (see Table 3.1). They may range from minor random fluctuations or noise, to sudden abrupt changes of the underlying data distribution, and are not mutually exclusive in real-word environments. From a perspective of any biometric system, changes may originate from phenomena that are either static or dynamic in nature. In addition, those changes can originate from *hidden contexts*, like variations of illumination conditions or pose of the individual which haven't been modeled in the system because of the limited representativeness of previously-observed reference samples.

This paper will focus on abrupt, gradual and recurring changes. Figure 3.2 illustrates these types of change (Kuncheva, 2008) as they may be observed over time for a concept in a 2 dimensional space (in this example, $\mathbf{a} = (a_1, a_2)$), assuming that it is observed at discrete time steps. It also shows the progression of a corresponding change detection measure.

In this paper, FRiVS is performed under semi- and uncontrolled capture conditions, and concept changes are observed in new reference ROI patterns that are sampled from the underlying data distribution. The refinement of previously-observed concepts (e.g., new ROIs are captured for a known face angle), corresponds to gradual changes (see Fig. 3.2(a)), and data corresponding to newly-observed concepts (e.g., new ROIs are captured under new illumination conditions), corresponds to abrupt changes (see Fig. 3.2(b)). In addition, a new concept

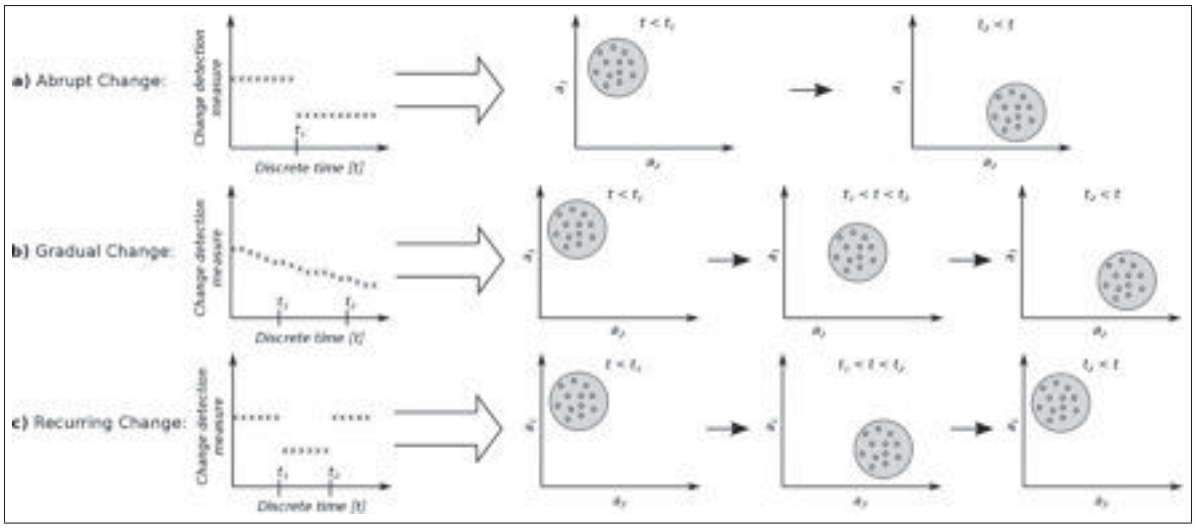


Figure 3.2 Illustration of (a) abrupt, (b) gradual and (c) recurring changes occurring to a single concept over time. The first column presents an example of the evolution of values of a change detection measure, corresponding to variations to the 2-D data distribution to the right.

(e.g., faces captured under natural vs. artificial lighting, or over a different face angle) can also correspond to a recurring change as specific observation conditions may be re-encountered in the future (see Fig. 3.2(c)). The rest of this section presents an overview of the different measures proposed in literature to detect changes, in order too choose the most adapted method for the proposed system. Then, specialized techniques that adapt classification systems to concept changes are reviewed, to introduce the proposed update strategies. Finally, a synthetic test case shows the benefit of using a change detection mechanism to guide the adaptation strategy used by the classification system according to the types of changes.

3.3.1 Detecting changes

In order to observe the occurrences of changes in the underlying data distribution, several families of measures have been proposed in the literature, which can be organised into techniques based on signal processing and pattern recognition. Prior to feature extraction, signal quality measures have been used to accept, reject, or reacquire biometric samples, as well as to select a biometric modality, algorithm, and/or system parameters (Sellahewa *et al.*, 2010). In an

FRiVS application, change detection can be performed by monitoring the values of an image-based quality over time. For example, several standards have been proposed to evaluate facial quality, such as ICAO 9303 (Doc, 2005), which cover image and face specific qualities. Other face quality measures compare input ROIs against facial references to assess image variations or distortions.

Change can also be measured after the feature extraction process of a biometric recognition system, and this paper focuses on pattern recognition techniques that rely on the feature distribution space, since the change detection process is designed to adapt the learning strategy of the classification module. These techniques fall into two categories: those that exploit classifier performance, and density estimation. Note that the accuracy of these measures depends heavily on the feature extraction and selection methods. Change detection mechanisms using classifier performance indicators have been considered for supervised learning applications (Kuncheva, 2004b). For instance, changes can be detected in system performance using accuracy, recall or precision measures on the input data (Gama *et al.*, 2004), or in the performance of a separate classifier dedicated to change detection, trained with the data corresponding to the last known change (Alippi *et al.*, 2011). However, while directly monitoring the system's performance is a straightforward way to measure concept changes, it can also have several drawbacks. Relying on a classifier's performance for change detection may require a considerable amount of representative training data, especially when a classifier must be updated (Alippi *et al.*, 2011). For this reason, the rest of this subsection will focus on density estimation measures and thresholding.

3.3.1.1 Density estimation measures

Although it may provide the most insight, detecting changes in the underlying distribution is very complex in the new data space. To reduce the computational complexity of change detection in the input feature space, several authors proposed to estimate the density of the data distribution. These techniques rely on fitting a statistical model to the previously-observed

data, which distribution in the input feature space is unknown, and then applying statistical inference tests to evaluate whether the recently-observed data belong to the same model.

As presented by Kuncheva (Kuncheva, 2009), clustering methods such as *k*-means or Gaussian Mixture Models (GMMs) may provide a compact representation of input data distributions in $\mathbb{R}^d$. In addition, Ditzler and Polikar (Ditzler and Polikar, 2011) and Dries and Ruckert (Dries and Rückert, 2009) proposed a non-parametric method that reduces the dimensionality of the incoming data blocks by representing them with feature histograms, with a fixed amount of bins. The following approaches have been proposed:

- Compute the **Likelihood** of the new data w.r.t. previously-generated model in order to quantify the probability that previous data blocks were sampled from the same concept. Kuncheva (Kuncheva, 2009) proposed to detect changes monitoring the likelihood of new data, using GMM or k-means to model the previous concepts.
- Monitor the **model parameters**, such as mean vectors and covariance matrixes of k-means and GMM models, in order to evaluate their relative evolution (Kuncheva, 2009), or polynomial regression parameters using the intersection of confidence interval rule, as proposed by Alippi et al. (Alippi *et al.*, 2011, 2013).
- Compare the estimated **densities** using measures like the Hellinger distances between consecutive histogram representation of data blocks (Ditzler and Polikar, 2011), or a binary distance, assigning a binary feature to each histogram bin and evaluating their respective coverage (Dries and Rückert, 2009).

Density estimation methods provide a lower level information than classifier performance indicators, and can therefore be more accurate for change detection. The performance indicators of classifiers trained over previously-encountered data are merely a consequence of possible changes in the underlying data distribution, while density estimation methods directly reflect the structure of underlying distributions. However, using a parametric estimation of density (e.g. GMM) makes strong assumptions concerning the underlying distribution of the input

data (Chandola *et al.*, 2009), and the amount of representative data and the selection of the method parameters are critical factors in accurate estimation of densities. For these reasons, non-parametric density methods based on histogram representation will be considered in the proposed system.

3.3.1.2 Thresholding

The detection of changes for a one-dimensional data has been extensively studied in the area of quality control for monitoring process quality (Kuncheva, 2009). Assuming a stream of objects with a known probability p of being defective (given from product specifications) several control chart schemes have been proposed. According to the basic Shewhart control chart scheme, a batch or window of samples of V objects are inspected at regular intervals. The number of defective objects is counted, and an estimate $\bar{p}$ is plotted on the chart. Using a threshold of $f\sigma$, where $\sigma = \sqrt{p(1-p)/V}$ and the typical value of $f = 3$, a change is detected if $\bar{p} > p + f\sigma$. Among the numerous control chart approaches, the popular CUmulative Sum (CUSUM) proposed to monitor the cumulative sum of classification errors at time t. Similarly, change detection in pattern recognition usually monitor one or several classifier performance indicators over time, to observe various patterns of change in a stream of input patterns, producing the decision through thesholding. For example, in (Klingenberg and Renz, 1998), the authors proposed a drift detection method to determine the optimal window size V containing the reference scores, which will be compared to the decision threshold. In the same way, the Hellinger Distance Drift Detection Method (HDDDM) method (Ditzler and Polikar, 2011) proposes to reset the data distribution using density estimation of the current data block if the change is detected. As with Klinkenberg and Renz (Klingenberg and Renz, 1998) and Gama et al. (Gama *et al.*, 2004), this method considers a growing window of samples (or data blocks), which reduces itself to the current data when a change is detected. In addition, decision is based on an adaptive threshold set on the previous values, adapting the final decision to the specific problem.

Given the changes that can occur in a FRiVS environment (and be observed from a set of ROI patterns extracted from a reference video), the system proposed in this paper will consider adaptive thresholding methods. In this case, decisions are based on the current capture conditions.

3.3.2 Adaptive classification for changing concepts

In the context of FRiVS, learning new reference ROI samples over time can raise the issue of preserving past knowledge, especially when new reference samples corresponding to new unknown concepts become available. Two categories of approaches have been proposed for supervised incremental learning of new concept in pattern recognition (Kuncheva, 2004b):

a. updating a single incremental classifier, where new reference data are assimilated after their initial training;

b. adding or updating one or more classifiers to an ensemble trained with the new data.

Several monolithic classifiers have been proposed for supervised incremental learning of new labeled data, providing mechanisms to maintain an accurate and up-to-date class model (Connolly *et al.*, 2008). For example, the ARTMAP (Carpenter *et al.*, 1992) and Growing Self-Organizing (Fritzke, 1996) families of neural network classifiers have been designed with the inherent ability to perform incremental learning. Other popular classifiers such as the Support Vector Machine (Ruping, 2001), the Multi-Layer Perceptron (Chakraborty and Pal, 2003) and Radial Basis Function neural networks (Okamato *et al.*, 2003) have been adapted to perform incremental learning. However, these classifiers are typically designed under the assumption that data is sampled from a static environment, where class distributions remain unchanged over time.

Recently, Connolly et al. (Connolly *et al.*, 2012) proposed a Dynamic Particle Swarm Optimization (DPSO) based incremental learning strategy allowing to optimize and evolve all parameters of an ARTMAP neural network classifier, performing incremental learning an pursu-

ing the optimization process to adapt to newly available data. However knowledge corruption is an issue with monolithic classifiers (Polikar *et al.*, 2001). Incremental learning of significantly different and noisy data can degrade the previously-acquired knowledge. For example, with ARTMAP networks, learning such data can lead to a proliferation of category neurons on the hidden layer, causing a reduction in discrimination for older concepts and an increased computational complexity. As highlighted by the *plasticity-stability* dilemma (Grossberg, 1988), a classifier should remain stable w.r.t. previously-learned concepts, yet allow for adaptation w.r.t. relevant new concepts that emerge in new reference data.

In contrast, adaptive ensemble methods have been proposed, combining diversified classifiers into an ensemble to improve the system's overall performance and plasticity to new reference data. They can be divided into three general categories (Kuncheva, 2004a):

a. *horse racing* methods, which train monolithic classifiers beforehand, and only adapt the combination rule dynamically (Blum, 1997; Zhu *et al.*, 2004);

b. methods using new data to update the parameters of ensemble's classifiers, in an online-learning fashion, like in (Gama *et al.*, 2004). In addition, Connolly et al. (Connolly *et al.*, 2013) proposed a DPSO-based incremental learning strategy to maintain an ensemble of optimized ARTMAP (Carpenter *et al.*, 1992) classifiers.

c. hybrid approaches, adding new base classifiers as well as adapting the fusion rule, such as the Learn++ algorithm (Polikar *et al.*, 2001), based on the popular Adaboost (Freund and Schapire, 1996), incrementally generates new classifiers for every new block of reference samples, and combines classifiers using weighted majority voting, the weights depending on the average normalized error computed during the generation process.

First of all, *horse racing* approaches cannot accommodate to new reference data since the classifiers in the ensemble are fixed, only the fusion rule changes. In addition, ensembles formed by online learners suffers from the same knowledge corruption issues than monolithic incremental classifiers. For example, in (Connolly *et al.*, 2013), the ARTMAP classifiers of the

MCS updated with new reference data over time are subject to knowledge corruption, as with the monolithic architectures using such classifiers. However, hybrid approaches provide a compromise between *stability* and *plasticity* to new data. Classifiers trained on previously acquired data, remains intact, while new classifiers are trained for the new reference data. For example, using the Learn++ algorithm (Polikar *et al.*, 2001), an ensemble is incrementally grown using, at each iteration, a weight distribution giving more importance to reference samples previously mis-classified, thus generating new classifiers specialized on the most difficult samples. Those systems may avoid knowledge corruption, but at the expense of growing system complexity, as new classifiers or reference samples are added to the ensemble for every new block of data. In addition, the update of the fusion rule tends to favor more recent concepts, as the weights of previously learned classifiers tend to decline.

More recently, approaches using a change detection mechanism to drive ensemble or incremental based adaptation strategies have been proposed. Minku et al. (Minku and Yao, 2012) proposed the Diversity for Dealing with Drifts algorithm, which maintains two ensembles with different diversity levels, one low and one high, in order to assimilate a new concept emerging in the observed data. When a significant change is detected though the monitoring of the system's error rate, the high diversity ensemble is used to assimilate new data and converge to a low diversity ensemble, and a new high diversity one is generated and maintained through bagging. Alippi et al. (Alippi *et al.*, 2013) also proposed a Just-in-Time classification algorithm, using a density-based change detection to regroup reference samples per detected concept, and update a on-line classifier using this knowledge when the observed data drift toward a known concept.

While these methods effectively rely on change detection and ensemble or incremental learning to adapt in changing environments, they emphasize newer concepts, through weighing or by discarding of the classifiers trained on previously-learned concepts. This can corrupt a FRiVS system's performance where every newer, and older and recurring concepts, are equally important.

In section 3.4, a new approach is proposed to adapt ensembles to new ROI reference patterns in a video-surveillance environment. It relies on the hypothesis that, when new reference data become available to adapt a facial models, and that the data incorporate an abrupt pattern of change w.r.t. to previously-learned concepts, previously-learned knowledge is better preserved with a learn-and combine strategy, instead of updating the previously-trained ones. As opposed to in literature, the resulting ensemble is specialized in every detected concept. The complexity of the system is controlled by the change detection mechanism, where a classifier is only added if significantly different reference data is presented, and knowledge of different concepts is updated over time when similar reference data are presented.

3.3.3 Synthetic test case: influence of changing concepts

A synthetic test case is now presented, to validate the intuition that, when new reference data incorporating abrupt changes w.r.t. previously-learned ones is presented to the system, it is more beneficial to employ a *learn-and-combine* strategy than updating previously-trained classifiers. This test case simulates a video person re-identification scenario: the FRiVS system operates in an environment where face captures may be sampled from different concepts (such as face orientation angle). This test case seeks to illustrate that when two significantly different sequences of data (abrupt change) are presented to a FRiVS system by the operator for update, training dedicated classifiers for each different concept provides better performance.

Consider a system designed to detect ROI samples from a target individual, among ROI samples from unknown non-taget individual. Two tagged data blocs, $Vs[1]$ and $Vs[2]$, are presented to the system, at time $t = 1$ (initial training) and the other at $t = 2$ (update during later operations), by the operator. Those blocks are comprised of reference patterns from the target and the non-target class, generated from two synthetic 2-dimensional sources, inspired from the rotating checkerboard classification problem (Kuncheva, 2004b) (Fig. 3.3), which provides samples distributed along a 2x2 checkerboard. In order to simulate the arrival of a new concept, $Vs[1]$ is composed of patterns from the initial checkerboard (Fig. 3.3(a)), and $Vs[2]$ from the checkerboard rotated by an angle of $\pi/4$ (Fig. 3.3(b)). At $t = 2$, the introduc-

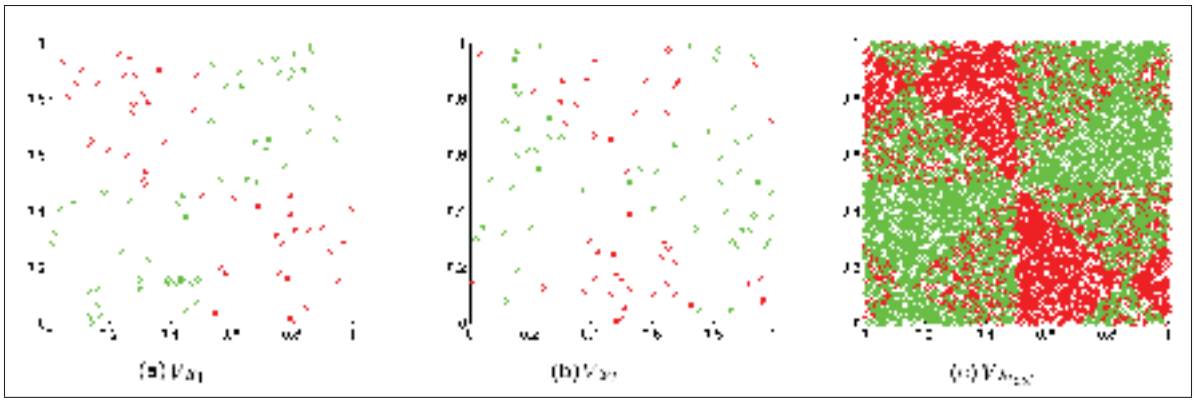


Figure 3.3 Reference and operational video sequences for the synthetic test case (Kuncheva, 2004b). Target class samples are represented in gray, and non-target ones in black.

tion of $Vs[2]$ represents an abrupt change w.r.t. to the patterns from $Vs[1]$. These are sampled from a different concept than the one modeled by the system at $t = 1$.

The operational mode is simulated by a combination of test blocks Vs_{test}, composed by target and non-target patterns originating from both concepts (Fig. 3.3(c)). As $Vs[1]$ and $Vs[2]$ incorporate data corresponding to two different concepts present in Vs_{test}, the update of the system with $Vs[2]$ should not corrupt previously-acquired knowledge, as it also corresponds to relevant information about Vs_{test}. This simulates a FRiVS scenario, where the operator gradually present the systems with tagged reference video sequences containing data from different concepts, e.g. different observation conditions such as camera angle, that are equally important in the system's operation, as future input patterns (ROIs) the system will capture in operations can correspond to any of those concept (face angle).

Two different training strategies are compared. With the *incremental* strategy, $Vs[1]$ and then $Vs[2]$ are learned incrementally by a Probabilistic Fuzzy ARTMAP (PFAM) (Lim and Harrison, 1995) classifier. With the *learn and combine* strategy, $Vs[1]$ and $Vs[2]$ are learned by two different PFAM classifiers, forming an ensemble which output is combined by the *average* score fusion rule. The *learn and combine* strategy is an implementation of the proposed

approach, by assuming a perfect mechanism to detect an abrupt change (a new concept) with the sequence $Vs[2]$ at $t = 2$ w.r.t. to $Vs[1]$.

Each PFAM classifier is trained with standard hyper-parameters values $\mathbf{h} = (\alpha = 0.001, \beta = 1, \varepsilon = 0.001, \bar{\rho} = 0, r = 2)$. $Vs[1]$ and $Vs[2]$ are composed of 50 target and 50 non-target patterns, and Vs_{test} of 2000 target and 2000 non-target patterns originating from both concepts (1000 patterns for each concept). In addition, two validation blocks $Vv[1]$ and $Vv[2]$ of 25 target and 25 non-target patterns each are considered to select a threshold that respects the operational constraint of $far \leq 5\%$. The operating point is selected based on ROC curve produced by systems adapted using the incremental and learn and combine strategy over the two validation datasets, and the performance is measured in terms of partial AUC ($pAUC$) for fpr $\in [0, 00.05]$, fpr, tpr and F_1 measures for selected operating points. In addition, the complexity of the systems are evaluated by counting the sum of F_2 layer neurons (category prototypes) of the PFAM classifiers. As the dataset is randomly generated from the sources, the simulations have been repeated for 100 replications. Results presented in Table 3.2 are the average values and the standard deviation, computed using a Student distribution and a confidence interval of 10%.

Table 3.2 Performance for the rotating checkerboard data of a PFAM-based system updated through incremental learning and trhough the learn and combine strategy. In the latter case, the classifiers are fused using the average score-level rule. Arrow ↑ (↓) represents a measure that should be maximised (minimized). Performance measures are defined in section 3.5.4.

Performance measures	**Enrolment with data from $Vs[1]$ with a single classifier**	**Update with data from $Vs[2]$ with**	
		incremental	**L&C**
pAUC(5%)(↑)	7.2% ± 0.4	6.2% ± 0.6	**8.2**% ± 0.6
fpr(↓)	17.07% ± 1.06	**9.85%** ± 0.96	14.94% ± 1.02
tpr(↑)	37.72% ± 2.17	16.19% ± 1.61	**32.5%** ± 2.27
$\mathbf{F_1}$(↑)	46.41% ± 1.94	23.63% ± 2.01	**41.35%** ± 2.21
Complexity(↓)	6.14 ± 0.23	**13.1** ± 0.3	17.0 ± 0.42

As shown in Table 3.2, after updating classifiers with data from the second concept of $Vs[2]$, the $pAUC(5\%)$ of the *incremental* PFAM strategy declines slightly, which is a consequence of

knowledge corruption (due to the learning of new data exhibiting significant concept change). While the increase of complexity is lower for the *incremental* strategy, it can be noted that the number of prototypes doubles when the system is presented with data from $Vs[2]$. This proliferation is a consequence of the incremental learning of two significantly different blocks of data.

Both systems start with the same performance level (after training with $Vs[1]$), but the training for the second concept with the *learn and combine* strategy generates significant increase in performance in terms of *pAUC*, tpr and F_1. Although the fpr decreases more with the *incremental* strategy, the decline in tpr and F_1 is considerably lower compared to the *learn and combine* strategy. Overall, this synthetic test case shows the benefit of training new classifiers to learn from new reference data that exhibit significant (abrupt changes). When presented with data from $Vs[2]$, the *learn-and-combine* strategy enabled to increase the system's performance, while the *incremenal* strategy is unable to preserve previously acquired knowledge.

3.4 An Adaptive Multi-Classifier System with Change Detection

Figure 3.4 presents an Adaptive Multi-Classifier System with Change Detection ($AMCS_{CD}$) specialized for video-to-video FR, with a novel updating strategy based on change detection. The main intuition at the origin of this contribution is that, when new reference samples become available to adapt a facial model, and that the data incorporate an abrupt change compared to existing concepts in the system, it is more beneficial to design a new dedicated classifier on the data and combine it to previously-learned classifiers in an ensemble (learn-and combine strategy), instead of updating the previously-trained ones (incremental learning strategy).

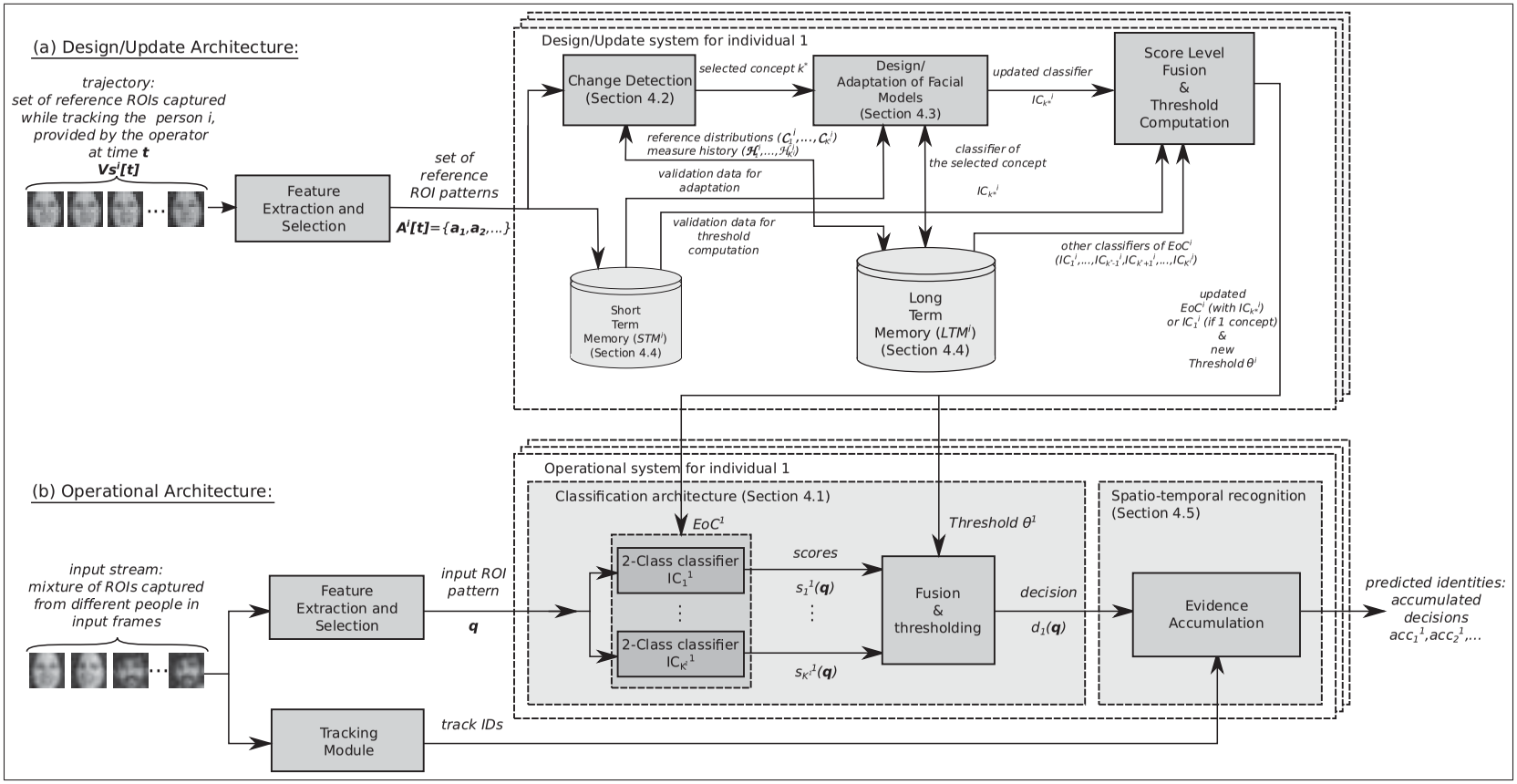


Figure 3.4 Architecture of the proposed $AMCS_{CD}$ for FRiVS. The design and update architecture for individual of interest i is presented in (a), and the operational architecture (for all individuals) in (b).

This enables to maintain an up-to-date representation of every concept encountered in the reference data, and avoid knowledge corruption when presented with new reference video encompassing new concepts in the feature space (e.g. face poses, illumination conditions,...).

For each individual $i = 1, ..., I$ enrolled to the system, this modular system is composed by an ensemble of K^i incremental two-class classifiers $EoC^i = \{IC^i_1, ..., IC^i_{K^i}\}$, where $K^i \geq 1$ is the number of concepts detected in the individual's reference videos, and a user specific threshold θ^i. The supervised learning of new reference sequences by the incremental classifiers is handled by a design and adaptation module, guided by change detection. For each individual, this module relies on long-term memory LTM^i to store the concept representations $\{\mathscr{C}^i_1, ..., \mathscr{C}^i_{K_i}\}$, and a short term memory STM^i to store reference data for design or adaptation and for validation.

Overall training/update process: The class-modular architecture for the proposed AMCS allows to design and update facial models independently for each individual of interest (see Alg. 18 and Fig. 3.4(a)). When a new reference video sequence $Vs^i[t]$ is provided by the operator at time t, relevant features are first extracted and selected from each ROI in order to produce the set of input patterns $\mathbf{A}^i[t]$ (Alg. 18, line 1). STM^i temporarily stores validation data used for classifier design and threshold selection ((Alg. 18, line 4). The change detection process assess whether the underlying data distribution of $\mathbf{A}^i[t]$ exhibits significant changes compared to previously-learned data. For this purpose, the previously-observed concepts $\{\mathscr{C}^i_1, ..., \mathscr{C}^i_{K^i}\}$ stored in LTM^i are compared to a histogram representation of $\mathbf{A}^i[t]$ (Alg. 18, lines 6-7). If a significant (abrupt) change (Fig. 1.2 and Table 3.1) is detected w.r.t. all the stored concept models, or if $Vs^i[t]$ is the first reference sequence for the individual (no previous concept has been stored), a new concept is assumed (Alg. 18, line 8). In this case, K^i is incremented, and a new incremental classifier $IC^i_{K^i}$ is designed for the concept (IC^i_1 if the first concept) and the user-specific threshold θ^i is updated (or created) using the training and adaptation module with the data from STM^i (Alg. 18, line 10 to 13). Note that the training of the classifier is done using non-target reference patterns (from other individuals) mixed to target reference patterns from $\mathbf{A}^i[t]$. When a moderate (gradual) change is detected, the classifier $IC^i_{k^*}$ corresponding

Algorithm 3.1: Strategy to design and update the facial model of individual i.

1 **Input:** *Sequence of reference ROIs for individual i* $Vs^i[t]$*, provided by the operator at time t.*;
2 **Output:** *Updated ensemble* EoC^i;
3 - Compute $\mathbf{A}^i[t]$, the set of reference ROI patterns obtained after feature extraction and selection of ROIs of $Vs^i[t]$;
4 - $STM^i \leftarrow \mathbf{A}^i[t]$;
5 **for** *each concept* $k \leftarrow 1$ **to** K^i **do**
6 - Measure $\delta_k^i[t]$ the distance between $\mathbf{A}^i[t]$ and the concept representation $\mathscr{C}_k^i$;
7 - Compare $\delta_k^i[t]$ to the change detection threshold $\beta_k^i[t]$ of the concept k;
8 **end**
9 **if** $\delta_k^i[t] > \beta_k^i[t]$ *for each concept* $k = 1,...,K_i$*, or* $K_i = 0$ **then**
10 `//An abrupt change is detected or no concepts are stored` - $K^i \leftarrow K^i + 1$;
11 - Set index of the chosen concept $k^* \leftarrow K^i$;
12 - Generate the concept representation $\mathscr{C}_{K^i}^i$ from $\mathbf{A}^i[t]$ and store in LTM^i;
13 - Initiate and train new classifier $IC_{K^i}^i$ and the user-specific threshold θ^i using (target and non-target) data from STM^i;
14 - Update $EoC^i \leftarrow \{EoC^i, IC_{K^i}^i\}$;
15 **else**
16 `//A moderate change has been detected` - Determine the index of the closest concept $k^* \leftarrow \min\{\delta_k^i[t] : k = 1,...,K^i\}$;
17 - Update the corresponding incremental classifier IC_{k^*} of EoC^i and the user-specific threshold θ^i using data from STM^i;
18 **end**

to the closest concept representation $\mathscr{C}_{k^*}^i$ is updated and evolved through incremental learning, and the user-specific threshold θ^i is updated as well (Alg. 18, lines 17 and 18). Finally, if several concepts are stored in the system, the EoC^i is updated to combine the most accurate classifiers of the known concepts: if a new concept has been detected, a new classifier $IC_{K^i}^i$ is added to EoC^i (Alg. 18, line 14), and if a known concept k^* is updated, the corresponding classifier $IC_{k^*}^i$ is updated (Alg. 18, line 18). If only one concept has been detected, a single classifier is assigned to the individual, $EoC^i = IC_1^i$.

Overall operational process: During operations, the AMCS functions according to Alg. 3.2 and Fig. 3.4(b). When a ROI is detected in a new area of the input scene, a face tracker is

Algorithm 3.2: Operational strategy for one individual i.

1 **Input:** *Stream of input ROIs of the observed individuals, ensemble of classifiers* EoC^i *for individual* i;
2 **Output:** *Accumulated decisions* $\{acc^i_1,...,acc^i_J\}$ *for the J tracks detected in the set of input ROIs.*;
3 **for** *each ROI* $r = 1,2,...$ **do**
4 - Perform feature extraction and selection to obtain input pattern $\mathbf{q}_r$;
5 - Determine the track ID $tr(\mathbf{q}_r)$;
6 **for** *each concept* $k \leftarrow 1$ **to** K^i **do**
7 - Compute the positive matching score for $\mathbf{q}_r$ with the k^{th} classifier $s^i_k(\mathbf{q}_r)$;
8 **end**
9 - Perform fusion of the K^i scores and apply user specific thresholds θ^i to obtain the ensemble decision $d^i(\mathbf{q}_r)$;
10 **end**
11 **for** *each detected track* $j \leftarrow 1$ **to** J **do**
12 **for** *each ROI* $r = 1,2,...$ **do**
13 **if** $tr(\boldsymbol{q}_r) = j$ **then**
14 - $acc^i_j \leftarrow acc^i_j + d^i(\mathbf{q}_r)$;
15 **end**
16 **end**
17 **end**

initiated with the ROI, assigning it a track ID number $j = 1,...,J$. Then, the tracker produces the same track ID number for that face in subsequent frames. An input stream is thus a mixture of ROIs from different people, each one is associated with a track ID number $j = 1,...,J$. In parallel, the system extracts and selects input ROI patterns $\mathbf{q}$ in the same way than the update process (Alg. 3.2, line 3). Each input $\mathbf{q}_r$ is associated with its track number $tr(\mathbf{q}_r) \in (1,...J)$. For each individual i enrolled to the system, the final decision from the EoC^i $d^i(\mathbf{q}_r)$ is computed from the independent scores $s^i_k(\mathbf{q}_r)$ ($k = 1,...,K^i$) of the classifiers (Alg. 3.2, line 7), fusing them in the score or decision level (Alg. 3.2, line 8) and applying user-specific thresholds θ^i. Finally, the identity predictions are generated through the accumulation of decisions per track using the track IDs: for each track $j = 1,...,J$, the decisions based on ROI patterns associated

with this ID are accumulated to output the final decision (Alg. 3.2, line 12) according to:

$$acc_j^i = \{ \sum_{\mathbf{q}_r \in inputstream} d^i(\mathbf{q}_r); tr(\mathbf{q}_r) = j \} \tag{3.1}$$

The rest of this section provides more details on the different modules inside the AMCS. For each module, a particular implementation is also described, in order to build a fully-functional system.

3.4.1 Classification architecture

In operational mode (Alg. 3.2), the classification system seeks to produce a binary decision $d^i(\mathbf{q}_r)$ in response to each input pattern $\mathbf{q}_r$ submitted to the system for each module i. If $d^i(\mathbf{q}_r) = 1$, the system has matched the facial capture $\mathbf{q}_r$ to the enrolled individual i. Module i is comprised of a single 2-class incremental classifier IC_1^i or an ensemble $EoC^i = \{IC_1^i, ..., IC_{K^i}^i\}$ per enrolled individual i, as well as a user-specific decision threshold θ^i. Usually, $\mathbf{q}_r$ is a pattern generated from an ROI sample extracted from a continuous video stream.

A specific implementation: The classification architecture is composed of IC_k^i that are 2-class Probabilistic Fuzzy ARTMAP (PFAM) (Lim and Harrison, 1995) incremental classifiers, where each one is trained using a balanced sets of references samples from the target individual (from trajectories) against a random selection of non-target data from an universal and cohort model (UM and CM). PFAM classifier is a versatile classifier that is known to provide a high level of accuracy with moderate time and memory complexity (Lim and Harrison, 1995). It is promising for face matching due to its ability to perform fast, stable, on-line, unsupervised or supervised, and incremental learning from limited amount of training data. Although trained for different concepts, the classifiers of every ensemble are designed using ROIs of the same individual, and can thus be considered as correlated. For this reason, following the recommendations in (Kittler and Alkoot, 2003), the score-level *average* fusion rule rule is used to combine the decisions in operational mode, producing the final ensemble's decision through the averaging of the classifier's scores. Finally, the authors have previously compared three classification

architectures for a FRiVS system (Pagano *et al.*, 2012): (1) a *global or monolithic architecture* composed of a single multi-class PFAM classifier, trained to detect the presence of all individuals of interest, (2) a *class-modular architecture* composed of a 2-class PFAM classifier per individual, and (3), a *class-modular architecture with ensembles of classifiers* composed of an ensemble of 2-class PFAM classifiers per individual. The latter was known to outperform other architectures when working with real video-based data.

The original fuzzy-ARTMAP classifier (Carpenter *et al.*, 1992) is composed by three layers: (1) the input layer $F1$ of $2D$ neurons (D being the dimensionality of the feature space), (2) a competitive layer $F2$ in which each of the N neuron corresponds to a category hyper-rectangle in the feature space, and (3), a map field of L output neurons (the number of classes, in that case $L = 2$). Connections between F_1 and F_2 are represented by a set of real-valued weights $\mathbf{W} = \{w_{dn} \in [0,1] : d = 1,2,...,D; n = 1,2,...,N\}$, and a category n is defined by a prototype vector $\boldsymbol{w}_n = (w_{1n}, w_{2n}, ..., w_{Dn})$. The F_2 layer is also connected to the F^{ab} layer through the binary-valued weight set $\boldsymbol{W}^{ab} = \{w^{ab}_{nl} \in 0,1 : n = 1,2,...,N; l = 1,2,...,L\}$. Vector $\boldsymbol{w}^{ab}_n = (w^{ab}_{n1}, w^{ab}_{n2}, ..., w^{ab}_{nL})$ represents the link between the F_2 category node n and one of the L F^{ab} class nodes. In supervised training mode, the synaptic weights are adjusted to the training patterns by (1) learning category hyper-rectangles in the feature space, and (2), associating them to the corresponding output classes. PFAM classifier (Lim and Harrison, 1995) relies on the fuzzy ART clustering and MAP field in order to approximate to the underlying data distribution as a mixture of Gaussian distributions in the feature space, and generates of prediction scores instead of binary decisions. In addition to FAM category hyper-rectangles and $F_2 - F^{ab}$ connexions, PFAM also learns prior probabilities $p(i)$ for each class i, categories center $\boldsymbol{w}^{ac}_n$ and covariance matrices Σ_n for each category n. PFAM dynamics are governed by a vector of five hyper-parameters $\mathbf{h} = (\alpha, \beta, \varepsilon, \bar{\rho}, r)$: the choice parameter $\alpha > 0$, the learning parameter $\beta \in [0,1]$, the match-tracking parameter $\varepsilon \in [-1,1]$, the vigilance parameter $\bar{\rho} \in [0,1]$, and the smoothing parameter $r > 0$.

As FAM or PFAM classifiers categorize the feature space into hyper-rectangles or Gaussian distributions (priors, centers and covariance matrices) during training, their memory complexity

and processing time in operations depend on the number of categories, or prototypes (Gaussian centers). The operational memory complexity of classification systems using PFAM classifiers can thus be compared based on the number of prototypes.

3.4.2 Change detection

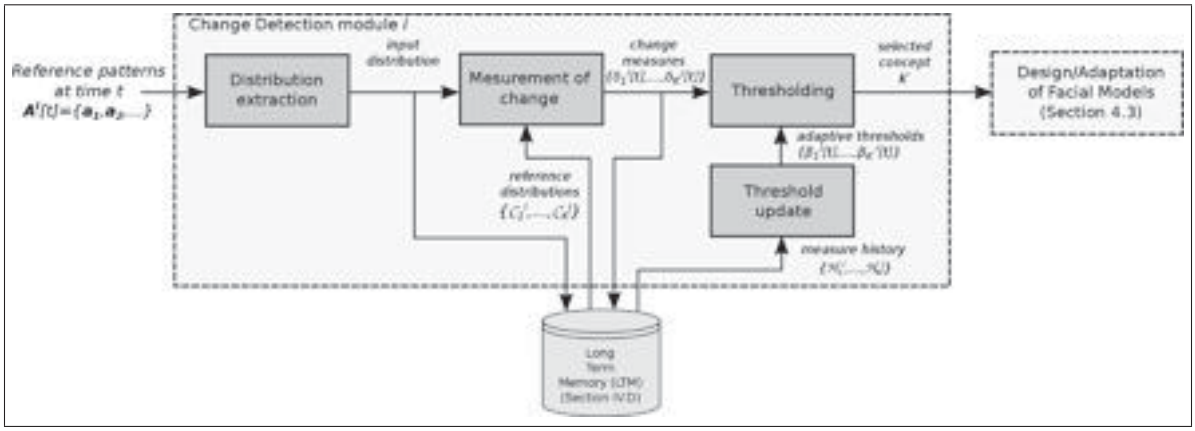


Figure 3.5 Architecture of the CD module i.

A change detection (CD) module (see Fig. 3.5) is proposed to distinguish abrupt from gradual changes that have emerged from the underlying distribution. It allows to trigger one of the strategies to adapt facial models in the AMCS. For each individual i, this module relies on a set of concept representations $\{\mathscr{C}_1^i, ..., \mathscr{C}_{K^i}^i\}$ and an history of distance measures $\{\mathscr{H}_1^i, ..., \mathscr{H}_{K^i}^i\}$ between all the previously-learned sequences of reference ROI patterns for individual i, and each concept representation. When a new reference pattern set $A^i[t]$ (extracted from a sequence $Vs^i[t]$) is presented to the system, the CD module detects if it differs significantly from previously-learned concepts. The input distribution $\mathscr{A}$ is extracted, and change is measured w.r.t. all stored concept representations $\{\mathscr{C}_1^i, ..., \mathscr{C}_{K^i}^i\}$. For each stored concept k, the measure $\delta_k^i[t]$ is compared to an adaptive threshold $\beta_k^i[t]$, computed from the measure history of the concept $\mathscr{H}_k^i$. The most appropriate concept k^* is then selected and provided to the adaptation module.

A specific implementation: Changes are detected using the HDDM presented in (Ditzler and Polikar, 2011), and the concepts are represented as histograms $\mathscr{C}_k^i$. This method provides a non-parametric low complexity detection measure though discetization of the feature space, which is a compromise between the precision of low level change detection of the density methods and the low complexity of the performance-based ones. In addition detections are based on the current contextual environment thanks to the adaptive threshold computation.

The HDDM-based (Ditzler and Polikar, 2011) CD process for each individual i is presented in Alg. 3.3. The reference sequence's histogram $\mathscr{A}$ is first computed from the patterns $\mathbf{A}^i[t]$, after feature extraction and selection (Alg. 3.3, line 3). Then, for each saved concept $k = 1, ..., K^i$, the Hellinger distance $\delta_k^i[t]$ is computed between histogram $\mathscr{A}$ and the concept representation $\mathscr{C}_k^i$ (Alg. 3.3, line 8), following:

$$\delta_k^i[t] = \frac{1}{D}\sum_{d=1}^{D}\sqrt{\sum_{b=1}^{B}\left(\sqrt{\frac{\mathscr{A}(b,d)}{\sum_{b'=1}^{B}\mathscr{A}(b',d)}} - \sqrt{\frac{\mathscr{C}_k^i(b,d)}{\sum_{b'=1}^{b}\mathscr{C}_k^i(b',d)}}\right)^2} \tag{3.2}$$

where D is the dimensionality of the feature space, B the number of bins in $\mathscr{A}$ and $\mathscr{C}_k^i$, $\mathscr{A}(b,d)$ and $\mathscr{C}_k^i(b,d)$ the frequency count in bin b of feature d. An abrupt change between the histogram $\mathscr{C}_k^i$ of concept k and $\mathscr{A}$ is detected if $\delta_k^i[t] > \beta_k^i[t]$, where $\beta_k^i[t]$ an adaptive threshold computed from the previous distance measures according to (Ditzler and Polikar, 2011):

$$\beta_k[t] = \hat{\mathscr{H}}_k^i + t_{\alpha/2}\cdot\frac{\hat{\sigma}}{\sqrt{\Delta_t}} \tag{3.3}$$

where α is the confidence interval of the t-statistic test, Δ_t the total amount of past distance measures stored in $\mathscr{H}_k^i$, and $\hat{\mathscr{H}}_k^i$ and $\hat{\sigma}$ the average and variance of those measures. If an abrupt change is detected for all the concepts (Alg. 3.3, line 13), or if $A^i[t]$ is the first sequence of reference ROI patterns provided for the individual i (Alg. 3.3, line 5), a new concept is added to the system. The number of concepts K^i for the individual i is incremented, and $\mathscr{A}$ is memorized into LTM^i as histogram $\mathscr{C}_{K^i}^i$ (Alg. 3.3, line 15).

Algorithm 3.3: Specific implementation of HDDM based CD for individual i.

1 **Input:** *Set of ROI patterns for individual i provided by the operator at time t,* $\boldsymbol{A}^i[t] = \{\boldsymbol{a}_1, \boldsymbol{a}_2, ...\}$;
2 **Output:** *Index* k^* *of the selected concept*;
3 - Generate histogram $\mathscr{A}$ of $\mathbf{A}^i[t]$;
4 **if** $K_i = 0$ **then**
5 - $newConcept \leftarrow true$;
6 **else**
7 **for** $k \leftarrow 1$ **to** K^i **do**
8 - $\delta_k^i[t] \leftarrow$ Hellinger distance between $\mathscr{A}$ and $\mathscr{C}_k^i$;
9 - Update threshold $\beta_k^i[t]$;
10 **if** $\delta_k^i[t]) \leq \beta_k^i[t]$ **then**
11 - $newConcept = newConcept \& false$;
12 **end**
13 **end**
14 **end**
15 **if** $newConcept == true$ **then**
16 - $K^i \leftarrow K^i + 1$;
17 - Store $\mathscr{C}_{K^i}^i \leftarrow \mathscr{A}$ into LTM^i;
18 `//Initialization of the measure history` $\mathscr{H}_{K^i}^i$
19 **for** $r \leftarrow nRep$ **do**
20 - Separate $\mathscr{C}_{K^i}^i$ into two sub-blocks $\mathbf{c}_1$ and $\mathbf{c}_2$ using the k-means algorithm;
21 - Compute $\delta_m(r)$, the Hellinger distance between $\mathbf{c}_1$ and $\mathbf{c}_2$;
22 **end**
23 - Re-organize measures $\{\delta_m(1), ..., \delta_m(nRep)\}$ in descending order;
24 - Initialize $\mathscr{H}_{K^i}^i \leftarrow \{\delta_m(1), \delta_m(2)\}$;
25 **else**
26 - Select the index of the concept to update $k^* = \min\{\delta_k^i[t]; \delta_k^i[t] \leq \beta_k^i[t], k = 1...K^i\}$;
27 - Update the concept model $\mathscr{C}_{k^*}^i \leftarrow \mathscr{C}_{k^*}^i + \mathscr{A}$;
28 - Update the measure history $\mathscr{H}_{k^*}^i \leftarrow \{\mathscr{H}_{k^*}^i, \delta_{k^*}^i[t]\}$;
29 **end**

The measure history $\mathscr{H}_{K^i}^i$ is initialized by: (1) separating of $\mathscr{A}$ into two sub-blocks $\mathbf{c}_1$ and $\mathbf{c}_2$ using the k-means algorithm (Alg. 3.3, line 18), and (2), computing the Hellinger distance δ_m between the 2 sub-blocks (Alg. 3.3, line 19). As the initialization of the k-means algorithm is random, this process is repeated for several replications $nRep$, and the 2 longest distances are stored in the concept's memory $\mathscr{H}_{K^i}^i$ (Alg. 3.3, line 20 and 21). The choice of the longest dis-

tances enables to generate a more permissive threshold for the subsequent reference sequences. It is considered as an estimation of the longest tolerable distance between reference sequences from the same concept. In addition, at least 2 measures must be selected in order to compute a proper variance for the next change detection.

If at least one comparison does not trigger CD, the closest reference histogram $\mathscr{C}^i_{k^*}$ (Alg. 3.3, line 23) is selected, and updated using $\mathscr{A}$ following $\mathscr{C}^i_{k^*} \leftarrow \mathscr{C}^i_{k^*} + \mathscr{A}$ (Alg. 3.3, line 41). The distance $\delta^i_{k^*}[t]$ is added into the concept's measure history $\mathscr{H}^i_{k^*}$ (Alg. 3.3, line 25).

This CD mechanism allows for selective windowing over the training data, as several reference distributions $\{\mathscr{C}^i_1, ..., \mathscr{C}^i_{K_i}\}$ are stored. In addition, each histogram representations of a distribution $\mathscr{C}^i_k$ is paired with an adaptive threshold $\beta^i_k[t]$ in order to adapt the decision for specific reference samples. Finally, this strategy can handle recurring concept changes if $A^i[t]$ is composed of data similar to a previously encountered concept k^*. In this case, only the corresponding classifier will be updated.

3.4.3 Design and adaptation of facial models

This module is dedicated to the design and update of incremental-learning classifiers IC^i_k limited to individual i. It relies on the last state (internal and hyper-parameters) of the previously-learned classifiers, reference target and non-target ROI patterns from STM^i as well as the output k^* of the CD module. If $k^* = K^i$, i.e. a new concept is detected, a new incremental-learning classifier $IC^i_{K^i}$ is initiated and trained on $A^i[t]$. Otherwise, the classifier $IC^i_{k^*}$ is updated incrementally with $A^i[t]$.

A specific implementation: A Dynamic Particle Swarm Optimization (DPSO) training strategy is employed to train and optimize the PFAM classifiers. This incremental learning strategy that evolves pools of incremental learning classifiers in the hyper-parameter space has been described and applied to adaptive FR systems in (Connolly *et al.*, 2012).

For each individual i, this module relies on a pool of PFAM classifiers $\mathscr{P}_1^k$ per concept k ($k = 1,...K^i$). Each pool consists of classifiers trained with reference ROI samples from concept k,to produce the best (global best) classifier IC_k^i. It may be combined in EoC^i with best classifiers from other concepts. This DPSO incremental-learning strategy allows to co-jointly optimize PFAM parameters (internal weights $\mathbf{W}$, $\mathbf{W}^{ab}$, $\mathbf{W}^{ac}$ and Σ , hyper-parameters $\mathbf{h}$, and architecture) of the classifiers in $\mathscr{P}_k^i$ such that the fitness function (classification accuracy) is maximized. The DPSO algorithm has been chosen for its convergence speed, and the DPSO training strategy has already been successfully applied in state-of-the art adaptive face recognition systems in video (Connolly *et al.*, 2012).

More precisely, PSO is a population based stochastic optimization technique inspired by the behaviour of a flock of birds (Eberhart and Kennedy, 1995). In this implementation, each particle of a swarm moving in the optimization space is defined by the five hyper-parameters $\mathbf{h} = (\alpha, \beta, \varepsilon, \bar{\rho}, r)$ of a PFAM classifier. The particles move in the optimization space according to two factors: (1) their *cognitive influence* (previous search experience), and (2), the *social influence* (other particles' experience,in a neighbourhood). At a discreet iteration τ, the position (hyper parameters) of each particle (classifier) $\mathbf{h}(\tau)$ changes according to its inertia and the *cognitive* and *social* influences following Eq. 3.4, with w_0, w_1 and w_2 the inertia, cognitive and social weights, and r_0, r_1 and r_2 random parameters.

$$\mathbf{h}(\tau) = r_0.w_0(\mathbf{h}(\tau) - \mathbf{h}(\tau-1)) + r_1.w_1(\mathbf{h}_{cog} - \mathbf{h}(\tau)) + r_2.w_2(\mathbf{h}_{soc} - \mathbf{h}(\tau)) \tag{3.4}$$

During optimization, each particle thus begins at its current location, then continues moving in the same direction it was going according to the inertia weight while being attracted by each source of influence:

- Its best known position $\mathbf{h}_{cog}$, the cognitive influence, also known as its memory.
- The best position of the swarm $\mathbf{h}_{soc}$, the social influence.

The best position is defined using a fitness function, which is, in this case, the classification performance over validation data stored in STM^i of the classifiers trained with training data, with the hyper-parameters corresponding to the positions of the particles. When new reference data become available, or if an abrupt change is detected, a new pool of classifiers $\mathscr{P}^i_{K^i}$ is initiated: the positions of the particles (the hyper-parameters of the PFAM classifiers) are randomly initialized in the optimization space, and the classifiers (their internal weigths $\mathbf{W}$, $\mathbf{W}^{ab}$, $\mathbf{W}^{ac}$ and Σ) are empty. On the other hand, if a gradual change is detected, previously-trained classifiers of pool $\mathscr{P}^i_{k^*}$ are updated through supervised incremental learning: their starting position (hyper-parameters) and internal weights are the final state of the previous optimization, when a similar concept had been encountered and learned.

Finally, in order to adapt to the optimization space according to gradual changes, and pursue the training of the classifiers after a previous optimization, the adaptation and training module is implemented with a dynamic variant of the PSO algorithm. The PSO algorithm has been adapted for dynamic optimization problems though 2 types of mechanisms to: (1) maintain the diversity in the optimization space through a modification of the social influence (such as (Nickabadi *et al.*, 2008a)), (2) increase the diversity in the optimization space after convergence when a change is detected in the objective function (using the memory of the particles) (such as (Blackwell *et al.*, 2004)). For this specific implementation, the DNPSO variant presented in (Nickabadi *et al.*, 2008a) is used. DNPSO maintains diversity within a pool $\mathscr{P}^i_k$ in the optimization space by: (1) relying on a local neighbourhood topology to generate *sub-swarms* of particles around *local bests* (which are the best particles in a local neighbourhood), (2) allowing the evolution of free particles (not in any subswarms) to explore the optimization space independently, and (3), reinitializing the free particles with low velocities. The social source of influence is determined within each sub-swarm. The choice of the DNPSO variant is motivated by the greater exploration of the optimization space through the generation of sub-swarms. This enables to consider all optima during the optimization process, instead of restarting at the convergence area when new reference data used for adaptation that exhibit a gradual change.

3.4.4 Short and long term memories

The long term memory LTM^i stores the different parameters and models necessary to pursue system training and detect changes when new reference samples become available for an individual i. On the other hand, the short term memory STM^i is not memorized from one training session to another, and serves as a temporary storage for reference validation samples.

A specific implementation: For each detected concept $k = 1, ..., K^i$, LTM^i stores:

a. A pool of 2-class PFAM classifiers $\mathscr{P}_k^i$. The hyper-parameter vector $\mathbf{h}$ as well as the PFAM's internal parameters ($\mathbf{W}$, $\mathbf{W}^{ab}$, $\mathbf{W}^{ac}$ and Σ). This pool is evolved and updated using the DNPSO incremental learning strategy (see section 3.4.3).

b. An histogram concept representation $\mathscr{C}_k^i$, with the frequency of bins defined by the reference patterns corresponding to the concept.

c. The history of past change detection measures $\mathscr{H}_k^i$, which stores the Hellinger distances computed between the histogram representation of the previously-acquired reference data and the concept k, in order to be able to compute the adaptive change detection threshold.

The data stored in STM^i is used to perform the optimization of the classifiers in the different pools $\{\mathscr{P}_1^i, ..., \mathscr{P}_{K^i}^i\}$, and choose the user specific threshold θ^i, for the classifier IC_1^i or the ensemble EoC^i (after *average* score-level fusion), according to false alarm specifications.

3.4.5 Spatio-temporal recognition – accumulation of responses

As shown in Fig. 3.1, systems for FRiVS typically rely on face detection tracking and classification. Fig. 3.4 is an example of a system that combines spatial and temporal computations into separate, but mutually interacting processing streams that cooperate for enhanced detection of individuals of interest. The general track-and-classify strategy has been shown to provide a high level of performance in video-based FR (Matta and Dugelay, 2009). Since classification and tracking co-occur in parallel, they can collaborate to improve overall face recognition.

During operations, face tracking follows the position and motion of different faces appearing in the scene. The objective of the tracker is to regroup ROIs that belong to a same person, and is defined by a high quality track, in order to provide a robust decision for each track through evidence accumulation.

A specific implementation: Fusion of responses from the ensembles and the tracker is accomplished via evidence accumulation, which emulates the brain process of working memory (Barry and Granger, 2007). For each initiated track j, for each individual i enrolled in the AMCS, and for each consecutive ROI $\mathbf{q}_r$ associated with this track, the dedicated ensemble EoC^i generates a binary decision $d^i(\mathbf{q}_r)$ (true, the individual is recognized, or false). The accumulated response is computed with a moving overlapping window of size V ROIs, following:

$$acc^i_j(r) = \sum_{u=r-V/2}^{r+V/2} d_(\mathbf{q}_u) \tag{3.5}$$

Then, the presence of the individual i in the track j can be confirmed if the accumulated response goes over a user-defined threshold Γ^i of a consecutive number of activations.

3.5 Experimental Methodology

The performance of the proposed AMCS is evaluated for the detection of individuals of interest with video captured in person re-identification applications. In particular, experiments focus the impact of employing a change detection mechanism (see Section 3.4.2) within the AMCS to drive the adaptation of facial models from new reference videos exhibiting various forms of concepts change. The objective of the experimental methodology is to validate our main hypothesis: it is beneficial to incorporate new data from different and abruptly changing concepts with a learn-and-combine strategy than with an incremental one.

3.5.1 Video-surveillance data

The Carnegie Mellon University Face In Action (FIA) face database (Goh *et al.*, 2005) is composed by 20-second videos capturing the faces of 221 participants in both indoor and outdoor scenario, each video mimicking a passport checking scenario. Videos have been captured with 6 Dragonfly Sony ICX424 cameras at a distance of 0.83m from the subjects, mounted on carts at three different horizontal angles (0° and ±72.6°), and with two different focal length (4 and 8mm) for each. Cameras have a VGA resolution of 640x480 pixels and capture 30 images per second. Data have been captured in three separate sessions of 20 seconds, at least one month apart. During the first session, 221 participants were present, 180 of whom returned for the second session, and 153 for the third. Only indoor sequences were considered in this paper.

3.5.1.1 Pre-processing

Figure 3.6 Examples of ROIs captured by the segmentation algorithm from the cameras array of 6 during the different sessions, for individuals with ID 21 and 110.

To extract the ROIs, segmentation has been performed using the OpenCV v2.0 implementation of the Viola-Jones face and eye detection algorithm (Viola and Jones, 2004), and the faces have been rotated to align the eyes in order to minimize intra-class variations (Gorodnichy, 2005a).

Then ROIs have been scaled to a common size of 70x70 pixels. Examples of ROIs captured for two individuals are shown in Fig. 3.6. Features have finally been extracted from ROIs with the Multi-Bloc Local Binary Pattern (LBP) (Ahonen *et al.*, 2006) algorithm for block sizes of 3x3, 5x5 and 9x9 pixels, concatenated with the grayscale pixel intensity values, and reduced to ROI patterns of $D = 32$ features using Principal Component Analysis.

For each one of the 3 sessions, and for each individual, the FIA dataset have been separated into 6 video subsets, according to the different cameras (left, right and frontal view, with 2 different focal length, 4 and 8 mm), resulting into the following sequences with notation:

- F_1 (F_2, F_3), L_1 (L_2, L_3) and R_1 (R_2, R_3): respectively the sequences composed by the samples from the Frontal, Left ($-72.6°$) and Right ($72.6°$) view of Session 1 (2,3), with a 4-mm focal length.

- Fz_1 (Fz_2, Fz_3), Lz_1 (Lz_2, Lz_3) and Rz_1 (Rz_2, Rz_3): the sequences composed by the same samples from the cameras with zoom, 8-mm focal length.

The average number of detected ROIs per individual is presented in Table 3.3. It can be noted that there are fewer ROIs for the right orientation than for other poses. This can be explained by the fact that the OpenCV Viola & Jones algorithm has only been trained for frontal and left orientations. Therefore, sequences for the the right facial orientation subset are not considered for experimental evaluation.

The individuals of interests have been selected among individuals appearing in all 3 sessions, as those with at least 30 ROIs for every frontal and left sequences. Of those, 10 individuals fulfil this requirement, individuals with IDs: 2, 21, 69, 72, 110, 147, 179, 190, 198 and 201. The remaining samples are mixed and separated into two Universal Model (UM) subsets: one half are used to generate the training UM, while the remaining consists in unknown UM classes appearing in test.

Table 3.3 Average number of ROI captured per person over 3 indoor sessions (s = 1,2,3) of the FIA database.

	ROIs per camera					
Orientations	$\mathbf{F}_s$	$\mathbf{FZ}_s$	$\mathbf{R}_s$	$\mathbf{RZ}_s$	$\mathbf{L}_s$	$\mathbf{LZ}_s$
Session 1	81 ± 4	131 ± 5	11 ± 1	23 ± 2	33 ± 2	40 ± 2
Session 2	88 ± 5	143 ± 7	11 ± 1	20 ± 2	34 ± 3	36 ± 3
Session 3	85 ± 6	141 ± 9	10 ± 1	20 ± 2	42 ± 4	39 ± 3

3.5.1.2 Simulation scenario

The following scenario is proposed to simulate video-to-video FR as seen in person re identification applications.

Table 3.4 Correspondence between the 9 reference video sequences used to adapt proposed AMCSs and the original *FIA* video sequences.

Time step t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Reference sequences	$Vs[1]$	$Vs[2]$	$Vs[3]$	$Vs[4]$	$Vs[5]$	$Vs[6]$	$Vs[7]$	$Vs[8]$	$Vs[9]$
Corresponding FIA sequence	Fz_1 (S_1)		Fz_2 (S_2)		Fz_3 (S_3)		Lz_1 (S_1)	Lz_2 (S_2)	Lz_3 (S_3)

Design and update of the face models: To simulate the role of the FRiVS operator providing the system with new reference sequences over time to update its facial models, the reference sequences of ROI patterns $Vs[t]$ are presented, after pre-processing, for every discrete time step $t = 1,2,...,9$. To avoid a possible bias due to the more numerous ROI detected from the frontal sessions, the original *FIA* frontal sequences have been separated into two subsequences, forming a total of 9 sequences, presented in Table 3.4.

Video sequences used for design are populated using the samples from the cameras with 8-mm focal length (sequences Fz_1, Fz_2, Fz_3, Lz_1, Lz_2 and Lz_3) in order to provide better face capture quality for learning samples. ROIs captured during 3 different sessions and orientations may be sampled from different concepts. The transition from sequence 6 to 7 represents most abrupt concept change in the reference samples, as it involves a change of camera angle. Changes

observed from one session to another, such as from sequences 2 to 3, 4 to 5, 7 to 8 and 8 to 9 depends on the individual. As faces are captured over intervals of several months, some abrupt changes can be detected, such as changes in hairstyle, make-up or facial hair. Finally, intra-session changes, from sequences 1 to 2, 3 to 4 and 5 to 6 represent more gradual changes since all sequences were captured with frontal cameras from the same sessions.

Operational evaluation: In order to present different facial captures that the one used for adaptation, only the cameras with 4-mm focal length (sequences F_1, F_2, F_3, L_1, L_2, L_3) are considered for operational evaluation. While the scaling normalizes every facial capture to a same size, the short focal length adds additional noise (lower quality ROIs), thus accounting for reference samples that do not necessarily originate from the observation environment in a real-life surveillance scenario.

For each time step $t = 1, 2, ..., 9$, the systems are evaluated after adaptation, simulating the arrival of different individuals one by one, at a security checkpoint at the airport. For each of the 3 sessions and 2 considered camera angles, they are presented with the ROI patterns of the corresponding sequences for each individual, one after the other. Evaluation is performed with input data from every session and camera angle for every time step. This simulates a FRiVS scenario where different concepts may be observed during operations, but where the reference videos are not available at the same time. Instead, they are gradually tagged and submitted to the system for adaptation. Every possible concept (face orientation, facial expression, illumination condition, etc.) present in the operational data, is presented to the systems over time.

3.5.2 Reference systems

For validation of the proposed proposed *AMCS* with change detection (called $AMCS_{CD}$), its performance is compared to the following systems that do not exploit change detection:

- **Incremental AMCS**, $AMCS_{incr}$: Instead of detecting a changes in concepts as $AMCS_{CD}$, a unique concept is considered. This simulates an $AMCS_{CD}$ which never detects any changes, and systematically adapts one single classifier. The system is only composed by a single classifier per individual of interest, and its parameters are updated incrementally when new reference sequences become available. This approach is an implementation of the adaptive classification system presented in (Connolly *et al.*, 2012).

- **Learn and combine AMCS**, $AMCS_{LC}$: This system doesn't include a change detection mechanism either, simulating an $AMCS_{CD}$ which always detect a change. For every new reference sequence, it systematically triggers the generation of a new concept in the system. It is composed of an ensemble of classifiers per individual, each classifier designed with a different reference sequence.

The comparison between $AMCS_{CD}$ and these two variants enable to evaluate the benefits of using change detection to govern the adaptation strategy. In addition, the proposed $AMCS_{CD}$ is compared the reference open-set TCM-kNN (Li and Wechsler, 2005) presented in Section 3.2.2. As the TCM-kNN is a global (non class-modular) classifier, $AMCS_{CD}$ is also compared to a reference class-modular system using probabilistic class-modular k-NN classifier, adapted to the FRiVS application, VSkNN. A separate k-NN classifier using Euclidean distance is considered for each individual of interest i, trained using positive reference samples from video sequences of target individual i, and a mixture of negative reference samples from the UM and CM, as with the other *AMCS*. A score is then computed through the *probabilistic kNN* approach (Holmes and Adams, 2002): the probability of the presence of the individual i is the proportion, among the k nearest neighbours, of reference samples from the same individual. The value of k is also validated through (2x5 folds) cross validation, along with the final decision threshold θ^i.

To improve the scalar performance of the proposed $AMCS_{CD}$ for the selected operating point in validation, a variant called $AMCS_w$ is also tested, where fusion of ensembles is performed at score level. It uses a weighted average to favour scores that are highest w.r.t. their threshold,

and filter out possible ambiguities. For an individual i with a concept-specific threshold θ_k^i (determined with validation data for concept k), and for each score $s_k^i(\mathbf{q})$, the weight ω_k^i is defined by the confidence measure $\omega_k^i = max(0,(s_k^i(\mathbf{q}) - \theta_k^i)$. This weight reflects the quality of the input pattern $\mathbf{q}$ in reference to concept k. The output score is then the result of the weighted average $\sum_{k=1}^{K^i} \omega_k^i.s_k^i$.

3.5.3 Experimental protocol

For each system, simulations follow a (2x5 fold) cross-validation process for 10 independent replications for each experiment, with pattern order randomization at the 5th replication. The full protocol is presented in Alg. 3.4. For each time step $t = 1,...,9$, and each individual $i = 1,...,I$, the design or update of the system is first performed. Change is first detected (Alg. 3.4 line 3), in order to determine the index of the concept k^* closest to the patterns in $A^i[t]$. In the case of $AMCS_{incr}$ ($AMCS_{LC}$), change detection is bypassed, and k^* is automatically set to 1 ($K^i + 1$). Dataset $dbLearn^i$ is then generated (Alg. 3.4 line 4), it is used to perform training and optimization of the PFAM networks. It remains unchanged for the two sets of five replications for the results to remain comparable, and is composed of reference patterns from $A^i[t]$, as well as twice the same amount of non target patterns equally selected from the UM dataset and CM^i. More precisely, selection of non-target patterns is achieved using the Condensed Nearest Neighbor (CNN) algorithm (Hart, 1968). The same amount of target and non-target patterns is selected using CNN, and combined with the same amount (picked at random) of patterns not selected by the algorithm. This enables to select non-target patterns that are close to the decision boundaries, as well as patterns that represent the center of mass of the non-target population. For each independent replication $rep = 1,...,10$, $dbLearn^i$ is divided into the following subsets (Alg. 3.4 line 8), based on the 2x5 cross-validation methodology:

- $dbTrain^i$ (2 folds): the training dataset used to design and update the parameters of PFAM networks.

- $dbVal_{ep}{}^{i}$ (1 fold): the first validation dataset, used to validate the number of PFAM training epochs (the amount of presentations of patterns from $dbTrain^{i}$ to the networks) during the PSO optimization.

- STM^{i} (2 folds): the second validation dataset.

Algorithm 3.4: Experimental protocol for performance evaluation.

for *Each time step,* $t \leftarrow 1$ **to** 9 **do**
  for *Each individual,* $i \leftarrow 1$ **to** 10 **do**
    - Perform change detection using $A^{i}[t]$ to the stored concept representations $\{\mathscr{C}^{i}_{1},...,\mathscr{C}^{i}_{K^{i}}\}$ to determine the closest concept index k^{*}. If a new concept is detected, following Alg. 3.3, $k^{*} = K^{i}+1$;
    - Generate $dbLearn^{i}$ dataset. Positive or target samples are selected from the pattern reference sequence $A^{i}[t]$, and relevant negative or non-target samples from UM^{i} and CM^{i} (from sequences corresponding to the same time stamp) are selected with the CNN method (Hart, 1968) ;
    for *each independent replication,* $rep \leftarrow 1$ **to** 10 **do**
      if $rep = 5$ **then**
        - Randomize samples order in $dbLearn^{i}$;
      end
      - Separate $dbLearn^{i}$ into $dbTrain^{i}$, $dbVal^{i}_{ep}$, STM^{i};
      - Randomly separate STM^{i} into $dbValPSO^{i}_{1}$ and $dbValPSO^{i}_{2}$;
      - Adapt the PFAM network pool corresponding to concept k^{*}, $\mathscr{P}^{i}_{k^{*}}$, with the DNPSO training strategy, with $dbVal^{i}_{ep}$ for the stopping criterion of the training epochs of the PFAM classifiers, $dbValPSO^{i}_{1}$ to compute particles' fitness and $dbValPSO^{i}_{2}$ to select the global best classifier $IC^{i}_{k^{*}}$;
      - Assemble $EoC^{i} = \{IC^{i}_{1},...,IC^{i}_{K^{i}}\}$ with the updated (or new) $IC^{i}_{k^{*}}$;
      - Select the threshold θ^{i} (operating point) corresponding to a far of 5% in validation from the ROC curve produced by EoC^{i} presented with data from STM^{i};
      for *each operational pattern sequence* **do**
        - Computation of the independent, frame by frame, decisions (transnational analysis);
        - Accumulation of the decisions of the sequence (time analysis);
      end
    end
  end
end

STM^i is randomly divided into two PSO validation datasets $dbValPSO_1^i$ and $dbValPSO_2^i$ (Alg. 3.4 line 9). Then, the pool of classifiers $\mathscr{P}_{k^*}^i$ corresponding to the selected concept k^* are trained through the DNPSO learning strategy (Connolly *et al.*, 2012) (Alg. 3.4 line 10), using the following parameters: 60 particles per swarm; max of 30 iterations; neighborhoods of 6 particles; max of 40 sub-swarms; max of 5 particles per sub-swam; early stopping if the best solution ever encountered remains fixed for 5 iterations. The bounds for PFAM parameters during optimization are: $0 \leq \bar{\rho} < 1$; $0 \leq \alpha \leq 1$; $0 \leq \beta \leq 1$; $-1 \leq \varepsilon \leq 1$; $0.0001 \leq r \leq 200$. The fitness computation follows three steps: (1) the training dataset $dbTrain^i$ is presented to the PFAM network, and its performance is evaluated with $dbVal1^i$ – to avoid over-training, this step is repeated for several epochs until the performance converges or decreases, and the stopping criterion is that performance does not increase for two consecutive epochs, (2) the fitness function is evaluated using $dbValPSO_1^i$, and (3), the best particles are determined using the second validation dataset, $dbValPSO_2^i$, and are stored in a archive for each iteration of the optimization. This methodology has been proposed in (Dos Santos *et al.*, 2009) in order to overcome over-fitting through the selection of particles with the best generalization performance.

When an previously-learned concept is updated, an existing pool $\mathscr{P}_{k^*}^i$ is be evolved through this DNPSO incremental-learning strategy. The optimization resumes from the last state – each classifier of the pool keeps its previous state (network parameters), and incrementally learns the new data. On the other hand, when a significant change is detected, the proposed $AMCS_{CD}$ generates a new pool that is optimized for the new concept $\mathscr{C}_{K^i}^i$. The classifiers from each concept are then combined into $EoC^i = \{IC_1^i, ..., IC_{K^i}^i\}$, and a validation ROC curve is generated, characterizing the performance of EoC^i over all the samples of STM^i (Alg. 3.4 lines 11 and 12). The threshold θ^i corresponding to a fpr $\leq 5\%$ is stored for the evaluation of the operational performances of the system. Finally, patterns from the operational sequences (sequences from F_1, F_2, F_3, L_1, L_2, L_3, and for every individual in the dataset, see section 3.5.1.2) are presented to the systems one sequence at a time. Individual predictions are generated to evaluate the transaction (ROI match) level performance of EoC^i (Alg. 3.4 line 14). Then, EoC^i predictions

are accumulated over time according to a trajectory of individuals appearing in a scene. This time analysis allows to evaluate the complete system performance (Alg. 3.4 line 15).

3.5.4 Performance measures

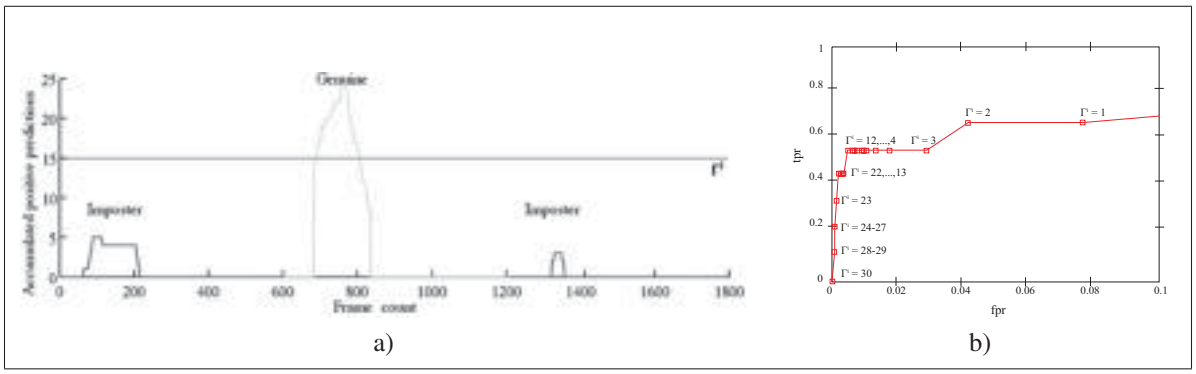
Transaction-level performance: Given the responses of a detector (or the final decision of an EoC) for a set of test samples, the true positive rate (tpr) is the proportion of positives correctly classified over the total number of positive samples. The false positive rate (fpr) is the proportion of negatives incorrectly classified (as positives) over the total number of negative samples. A ROC curve is a parametric curve in which the tpr is plotted against the fpr. In practice, an empirical ROC curve is obtained by connecting the observed (tpr, fpr) pairs of a soft detector at each threshold. The area under the ROC curve (AUC) or the partial AUC (for a range of fpr values) has been largely suggested as a robust scalar summary of 1- or 2-class classification performance. The AUC assesses ranking in terms of class separation – the fraction of positive–negative pairs that are ranked correctly. For instance, with an $AUC = 1$, all positives are ranked higher than negatives indicating a perfect discrimination between classes. A random classifier has an $AUC = 0.5$, and both classes are ranked at random. To focus on a specific part of the ROC curve, the partial AUC $pAUC$ can also be computed, as the partial area for a fpr less or equal to a specified value.

In a video-surveillance application, non-target individuals are often much greater than the target ones. ROC measure may be inadequate as it becomes biased towards the negative class (Weiss, 2003). For this reason, the precision-recall space has been proposed to remain sensitive to this bias. Indeed, the precision is defined as the ratio $TP/(TP+FP)$ (with TP and FP the number of true and false positives), and the recall is an another denomination of the tpr. Precision allows to assess the accuracy for target patterns. The precision and recall measures can be summarized by the F_1 scalar measure, which can be interpreted as the harmonic mean of precision and recall. Finally, a classifier can also be characterized by its *precision-recall* operating characteristics (P-ROC) curve, and the area under the P-ROC curve ($AUPROC$) can be considered as a robust performance measure.

Therefore, considering each ROI match independently, the systems' transaction-level performance will be assessed using:

- Local measures: tpr, fpr, *precision* and F_1. Those measures are specific to the operating point (threshold Θ^i), determined during system design.
- Global measures: *AUC*, *pAUC* and *AUPROC*. Those measures are a more general evaluation of the systems performance over the entire range of the possible operating points.

Performance of the full system over time: To evaluate the performance of the entire system proposed in this paper, individual-specific predictions of each ensemble are accumulated over a trajectory for robust decisions. More precisely, for each individual, the predictions are accumulated with a moving window of $V = 30$ ROIs in a trajectory. The individual is detected when the accumulated activations go past a defined threshold Γ^i.



a) b)

Figure 3.7 As an example, assume that individual 21 is enrolled to the $AMCS_{CD}$. After training sequence 9 (*Lz3*), the number of positive predictions accumulated over a fixed-size time window in presented in (a). Three sequences of 600 frames, from 3 different individuals, have been concatenated, with first a sequence from an impostor (in black), then from the genuine individual (21, in gray), and then from another impostor (in black). In (b), the overall accumulation ROC curve characterizing the $AMCS_{CD}$ performance for individual 21 over all the test sequences.

An example is presented in Fig. 3.7 (a), where 3 sequences of 600 frames have been concatenated. The first and the last one correspond to unknown individuals in the UM, while the

second one correspond to target individual 21. The predictions are generated by $AMCS_{CD}$ after the 9th training sequence (session Lz_3), dedicated to the individual 21. It can be observed that genuine predictions go significantly higher than the impostor's.

To assess the overall performance of the different systems for every individual i, an overall accumulation ROC curve is generated, with threshold Γ^i going from 0 to 30 (the size of the moving window). For each target sequence, a true positive occurs when the maximum value of the accumulated predictions goes over Γ^i. In the same way, a false positive occurs when the maximum value of the accumulated predictions for non-target sequences goes over the threshold. An example is presented in Fig. 3.7 (b). To summarize the system performances, the AUC of the overall accumulated ROC curves is used as with the transaction-level measures.

3.5.5 Memory complexity measures

The systems complexity is evaluated in operational mode, in order to compare resources required to predict the identity associated to an input ROI pattern.

As mentioned in Section 3.4.1, a PFAM network operational behaviour is one of a GMM, where cluster centres are the prototypes in the $F2$ layer. For each input ROI pattern, the final score is computed from the likelihoods of the different clusters. As a consequence, the memory and time complexity required to classify a facial ROI in operations is proportional to the number of prototypes in PFAM networks. For this reason, the operational memory complexity of *AMCS* systems will be compared based on the sum of the number of $F2$ layer neurons for all the PFAM classifiers in the ensembles.

Similarly, TCM-kNN and VSkNN both rely on a kNN classifier. For each input ROI pattern, an euclidean distance is computed for each reference pattern stored for kNN classifier. In VSkNN, those distances are then ordered to compute probabilistic scores as presented in Section 3.5.2. TCM-kNN adds more computational complexity, as the score computation relies on strangeness measures for each input ROI, requiring additional re-orderings. The operational

memory complexity of the identity prediction from an input ROI is thus also proportional to the number of reference patterns stored in the system, which will be used for comparison.

3.6 Results and Discussion

3.6.1 Change detection performance

Table 3.5 Changes detected per individual of interest (marked as a X) for each update sequence.

Individual ID	**Update Sequences** (time step t)									**Total per individual**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
2	X				X			X		3
21	X				X		X		X	4
69	X		X			X	X			4
72	X		X				X			3
110	X		X		X		X			4
147	X		X		X		X			4
179	X		X		X			X		4
190	X				X		X			3
198	X				X		X			3
201	X		X		X		X		X	5
Total per sequence	10	0	6	0	8	1	8	2	2	

For each individual of interest, Table 3.5 presents the update sequences for which changes have been detected, as well as the total number of detections. The first sequence corresponds to the initialization of the first concepts of each individual. The maximum number of detection for a sequence of 10, meaning that a change is detected for every individual. The 3 highest detection counts occur for the sequences 3, 5 and 7, and for 6, 8 and 8 of the individuals, respectively. These changes correspond to the introduction of training samples from the 2nd frontal session, the 3rd, and the 1st left session. Although the apparition of changes depends on the specific individuals (haircut change, hat, glasses, etc.), this result is expected since those 3 sessions are the most likely to exhibit significant abrupt changes: the two former occurred at least 2

and 3 months after the first update sequence, and the latter is the first introduction of samples captured from a different angle.

For a more detailed analysis, individuals 21 and 110 were considered. Changes detected in these cases can be correlated with ROIs shown in Fig. 3.6. For the individual 21, abrupt changes have been detected for update sequences 5 (introduction of patterns Fz_3), 7 (Lz_1) and 9 (Lz_3). As shown in Fig 3.6, the changes detected with the introduction of sequences 5 and 9 correspond to a change in make-up and hair style, while the change detected with sequence 7 is the introduction of left oriented samples. Similarly, as shown in Fig. 3.6, changes for individual 110 have been detected with sequence 3 (Fz_2), corresponding to hair-style and skin tone change, 5 (Fz_3), also corresponding to skin tone change and 7 (Lz_1), which is the introduction of the samples with left camera angle.

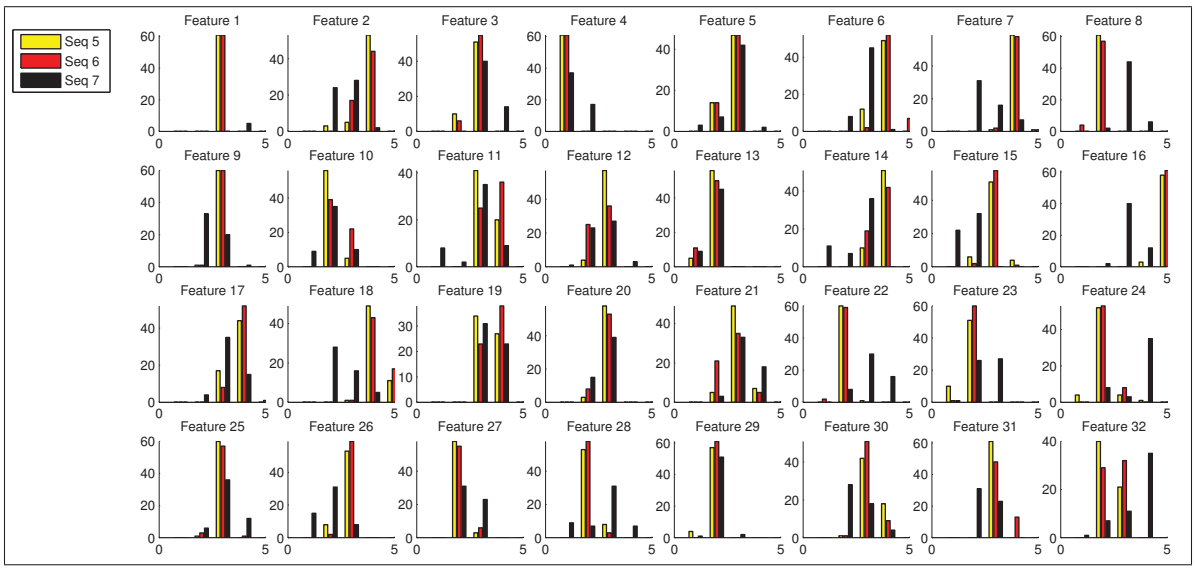


Figure 3.8 Histograms representation of the 5th, 6th and 7th sequence of patterns for the individual 21, in the feature space of input ROI patterns ($D = 32$ dimensions). The Hellinger distances between the sequence 5 and 6, and between 6 and 7 are respectively 0.0253 and 0.1119.

The abrupt change detected with sequence 7 can also be observed in the feature space, as illustrated by the significant differences in histogram representations of the sequences 5, 6

and 7 (see Fig. 3.8). Sequence 7 is visibly different for most features from sequences 5 and 6 (which belong to the same enrolment session, Fz_3). This difference is also shown in the Hellinger distance between the sequences 6 and 7, which is significantly higher than between the sequences 5 and 6.

Results confirm that the change detection module proposed for $AMCS_{CD}$ (Fig 3.4) can efficiently detect abrupt concept changes in sequences of facial captures of the *FIA* dataset. In response to new reference video sequences, this module allows $AMCS_{CD}$ to adapt facial models according to different strategies, either incremental learning or *learn-and-combine*.

3.6.2 Transaction-level performance

Average results: Fig. 3.9 presents the average overall transaction-level performance of proposed and reference systems, for the 10 individuals of interest according to fpr, tpr and F_1 measures (Fig. 3.9 (a), (b) and (c)) at an operating point selected (during validation) to respect the constraint fpr $\leq 5\%$, and the global $AUPROC$ measure over all fpr values (Fig. 3.9 (d)). Performance is assessed on predictions for each ROI captured in test sequences, after the systems are updated on each adaptation sequence.

In Fig. 3.9 (d), $AMCS_{CD}$, $AMCS_{incr}$ and $AMCS_{LC}$ exhibit a significantly higher level of $AUPROC$ performance than VSkNN and TCM-kNN. After learning the 9th update sequence, VSkNN and TCM-kNN have an average $AUPROC$ of 0.57 ± 0.04, while $AMCS_{incr}$, $AMCS_{CD}$ and $AMCS_{LC}$ are respectively at 0.82 ± 0.03, 0.89 ± 0.02 and 0.91 ± 0.01. Performing a *Kruskal-Wallis* test for those three measures using a *p-value* of 0.1, indicates that $AMCS_{incr}$ performance is significantly lower than $AMCS_{CD}$ and $AMCS_{LC}$, which are comparable. In addition, while these 3 *AMCS* yield similar performance after the first 2 sequences, $AMCS_{CD}$ and $AMCS_{LC}$ improve their performance more significantly than $AMCS_{incr}$ when samples from session Fz_2 are integrated into the systems. Average $AUPROC$ performance goes from 0.75 ± 0.03 and 0.79 ± 0.02 to 0.81 ± 0.02 and 0.83 ± 0.02 for $AMCS_{CD}$ and $AMCS_{LC}$, while it only goes from 0.76 ± 0.02 to 0.78 ± 0.02 for $AMCS_{incr}$. Sequence Fz_3 represents the most abrupt

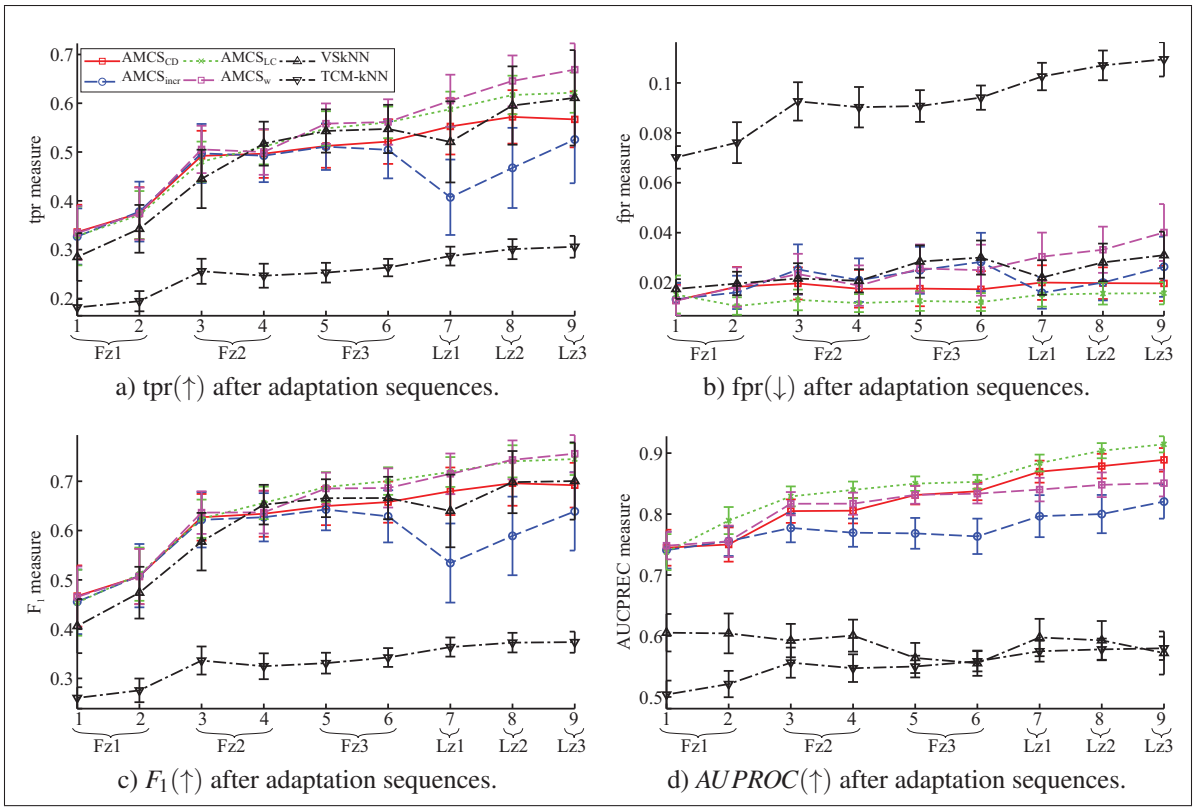


Figure 3.9 Average overall transaction-level performance of proposed and reference systems, after the integration of the 9 adaptation sequences. The average value of performance measures and confidence interval over 10 replications are averaged for the 10 individuals of interest.

changes for the frontal faces, captured several months later, along *left* pose, Lz_1, Lz_2 and Lz_3. The $AMCS_{CD}$ and $AMCS_{LC}$ benefit the most from learning this new data as their $AUPROC$ performance continues to diverge w.r.t. that of $AMCS_{incr}$ until the last update sequence.

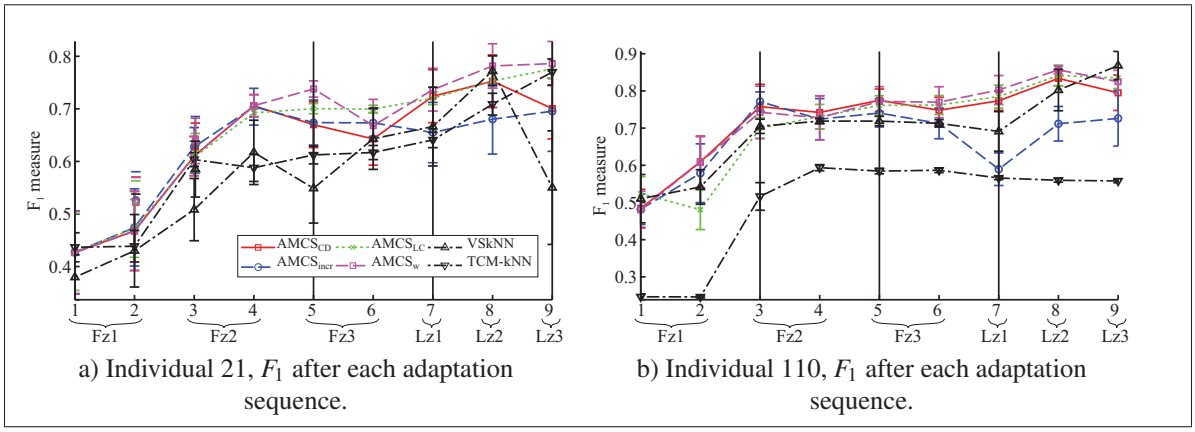
In terms fpr performance (Fig. 3.9 (b)) it can be first observed that all systems except TCM-kNN remain under the constraint of fpr $\leq 5\%$, with $AMCS_{LC}$ and $AMCS_{CD}$ significantly lower than $AMCS_{incr}$ and VSkNN. It can be noted that $AMCS_{CD}$ provides a lower fpr on test sequences. After learning update sequence 9, the average fpr for $AMCS_{CD}$ is at $1.97\% \pm 0.70$. On the other hand, the fpr of the $AMCS_w$ variant is more affected by the introduction of the new orientation after sequence 7, at it increases, after sequence 9, to $4.0\% \pm 1.13$. A closer reveals that false positives are mainly a consequence of an increase of score values for negative

samples for one of the classifiers in each ensemble. In most of the cases, when a change is detected and the fpr increases, most of the false positives are triggered by classifiers that correspond to newly-added concepts, not specialized to differentiate positive from negative samples of the other concepts. This produces a positive prediction for a non-target ROI from a different concept than the one of its training patterns. An increase of fpr can indeed be observed after learning from Lz_1 as the majority of changes (and thus the classifier addition) are detected at those transitions. Although always below the 5% constraint imposed in validation, $AMCS_w$ has the tendency to increase the false positives of different ensembles, as those scores are increased by the normalizations which set to zero other lower scores.

The F_1 measure (Fig. 3.9 (c)) gives a condensed view of the precision and recall (tpr) for the selected operating point, and it allows to observe the performance on target samples. The F_1 performances all systems except TCM-kNN are comparable until the update sequence 5. Updating on reference sequences from session Fz_3 enables $AMCS_w$ and $AMCS_{LC}$ to differentiate themselves, at respectively 0.69 ± 0.03 and 0.70 ± 0.03. However, the most significant decline in F_1 performance occurs for update sequence 7 (Lz_1 sessions), where $AMCS_{incr}$ performance decreases from 0.63 ± 0.05 to 0.53 ± 0.08, and the system requires two more sequences of *left* oriented captures to recover. The decrease of the F_1 performance is a consequence of a decrease in tpr, from $50.46\% \pm 5.9$ to $40.73\% \pm 7.7$ after learning sequence 7. This is a manifestation of the knowledge corruption that can occur in an incremental system, as the introduction of significantly different training patterns decreased its ability to effectively detect positive ones. $AMCS_{CD}$, $AMCS_w$, $AMCS_{LC}$ and VSkNN, on the other hand, do not suffer from the same effects, and their performance continues to improve over the last 3 update sequences. After learning the 9th update sequence, $AMCS_w$ and $AMCS_{LC}$ exhibit the highest level of F_1 performance at respectively 0.76 ± 0.04 and 0.75 ± 0.03. $AMCS_{CD}$ and VSkNN both end at about 0.70. The tpr boost induced by the fusion function of $AMCS_w$ provides the best F_1

performance, despite the higher fpr values. Finally, the performance of TCM-kNN remains significantly lower throughout the experiments.

Focus on individuals 21 and 110: Figure 3.10 presents average transaction-level F_1 performance obtained for individuals 21 and 110. They provide a bad (individual 21) and a good (individual 110) case for the proposed $AMCS_{CD}$. The F_1 performance of individual 110 leads



a) Individual 21, F_1 after each adaptation sequence.

b) Individual 110, F_1 after each adaptation sequence.

Figure 3.10 Average transaction-level performance after learning the 9 update sequences. Significant (abrupt) changes are indicated as vertical lines.

to similar observations as with the average overall evaluations: the performance declines in F_1 of $AMCS_{inc}$ at the 7th sequence while $AMCS_{CD}$ and $AMCS_{LC}$ continue to improve, and TCM-kNN performance remains below all the others. However, for individual 21, while the 7th sequence triggers a change detection, all systems exhibit similar performances and behaviour, without any decline in F_1 for $AMCS_{incr}$. With a closer examination of the videos, about 92% of the ROIs in the Lz_1 , Lz_2 and Lz_3 sequence for individual 110 are profile orientation, while the remainder are mostly 3/4 frontal views captured during a movement of the individual's head. In contrast only 51% of the ROIs of individual 21 correspond to a profile orientation, with a majority in the Lz_3 session (9th sequence), at the end of the simulation. Individual 21 can be considered as a case where the change detection process may be too sensitive - the new reference patterns provided in the sequences 7, 8 and 9 are not different enough for new classifiers

to have a considerable impact on transaction-level performance. Results for individual 110 shows that learning significantly diversified samples can be more effective with the proposed $AMCS_{CD}$. Finally, in both cases, $AMCS_w$ still exhibits similar F_1 performance to $AMCS_{LC}$.

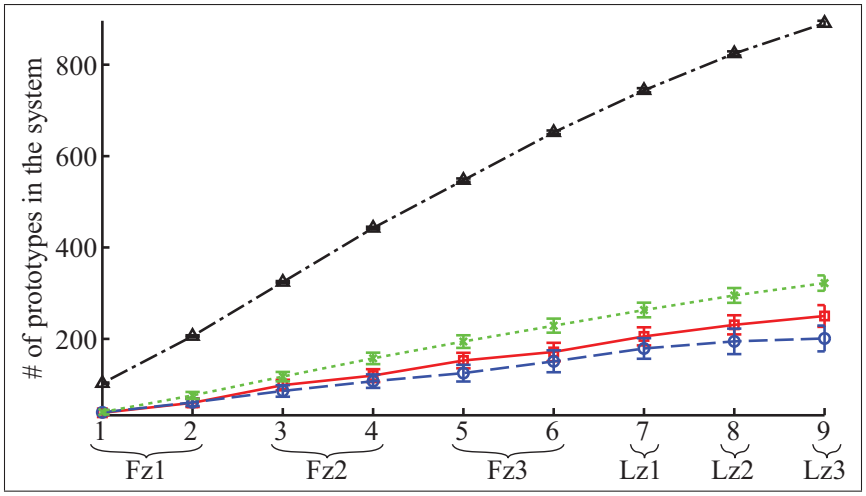


Figure 3.11 Average memory complexity. Amount of $F2$ prototypes for the *AMCS* systems, and amount of reference patterns for VSkNN and TCM-kNN, after learning of adaptation sequences. $AMCS_{CD}$ and $AMCS_w$ have the same amount of prototypes, as well as VSkNN and TCM-kNN.

Although the $AMCS_{CD}$ and VSkNN exhibits similar transaction-level performance, Fig. 3.11 shows that the amount of prototypes (sum of the number of $F2$ layer neurons for all the PFAM classifiers in an ensemble) needed by the the 3 *AMCS* is significantly lower than the number of reference patterns needed by VSkNN and TCM-kNN. The memory complexity of VSkNN and TCM-kNN grows to about 900 prototypes after the 9 adaptation sequences. The complexity of $AMCS_{CD}$ ($AMCS_w$), $AMCS_{incr}$ and $AMCS_{LC}$ remain comparable until the update sequence 5. Their sizes continue to grow until the last sequence, with $AMCS_{incr}$ the smaller system (200.84 ± 28.2), and $AMCS_{LC}$ the bigger one (322 ± 16.8). $AMCS_{CD}$ ends with 250 ± 13.7 prototypes. Considering that a prototype or reference sample weights 128 bytes (a vector of 32 $floats$ of 32 bits), the reference sample stored by VSkNN and TCM-kNN after the 9 adaptation sequences use up to 115 kB, while the prototypes of $AMCS_{CD}$ ($AMCS_w$), $AMCS_{incr}$ and $AMCS_{LC}$ respectively use around 32, 25.6 and 42.2 kB.

Overall, the proposed $AMCS_{CD}$ provides a compromise between the $AMCS_{incr}$ (low complexity but lower performance) and $AMCS_{LC}$ (significantly greater complexity but comparable performance). In this simulation, the $AMCS_{incr}$ exhibits the knowledge corruption problem, while the reference $AMCS_{CD}$ and VSkNN are prone to a increase in the system complexity. Those two problems have been presented in Section 3.3.2 as the main issues of the adaptive classification system in the literature. The proposed $AMCS_{CD}$ can achieve transaction-level performance comparable to the reference $AMCS_{LC}$ and VSkNN systems, but with a significantly lower computational complexity. In addition, $AMCS_{CD}$'s performance is significantly higher than the *open-set* TCM-kNN. By virtue of the change detection mechanism, it can also avoid the decline in performance due to knowledge corruption (seen with $AMCS_{incr}$) when learning significantly different adaptation sequences. Finally, although exhibiting higher fpr (but below the validation constraint) the $AMCS_{w}$ achieve significantly better performance in terms of tpr and similar F_1 than $AMCS_{CD}$, without being negatively affected by the introduction of different adaptation sequences as $AMCS_{CD}$.

3.6.3 Performance of the full system over time

In the proposed architecture (see Fig. 3.4), the face tracker groups ROIs corresponding to tacking trajectories initiated in each video sequence. Classification prediction for each ROI in each trajectory are accumulated over time. Considering that the transaction-level performance of the *open-set* TCM-kNN was consistently lower than the other systems, and that the system hasn't originally been designed to be used with an accumulation strategy, TCM-kNN's accumulation performance has not been evaluated.

Average results: The average accumulation performance are presented in Fig. 3.12. The accumulation performance of $AMCS_{CD}$, $AMCS_{LC}$ and $AMCS_{incr}$ are similar from sequence 1 to 6. VSkNN provides the best performance level for the two first sequences (0.84 ± 0.02 and 0.85 ± 0.02), and $AMCS_{w}$ exhibits similar performance to VSkNN from sequences 3 to 6. At sequence 6, $AMCS_{LC}$, $AMCS_{w}$, VSkNN exhibit accumulation performance comparable to $AMCS_{CD}$, $AMCS_{incr}$ and $AMCS_{LC}$. Then, it can be seen that the accumulation process filters

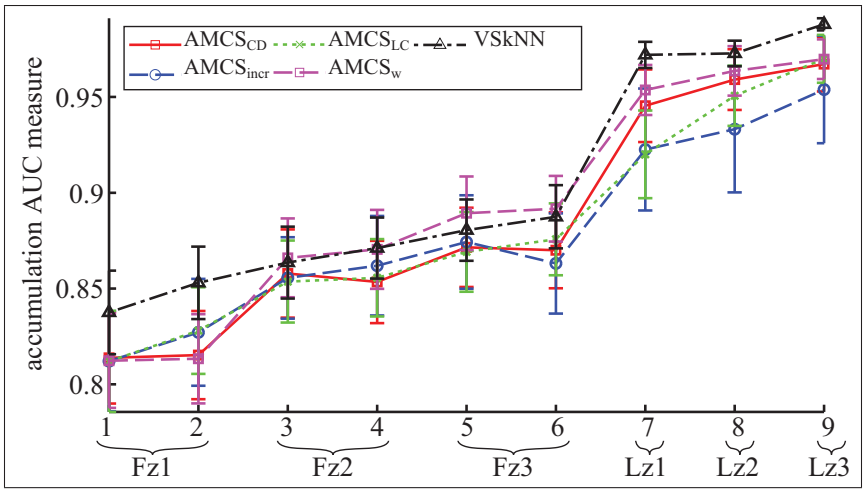


Figure 3.12 Average accumulation AUC performance after learning the 9 update sequences.

out the irregularities, and their accumulation performance increases after the introduction of sequences from Lz_1. The increase is however more important for VSkNN, $AMCS_w$ and $AMCS_{CD}$, which respectively go to 0.97 ± 0.01, 0.95 ± 0.01 and 0.95 ± 0.01. $AMCS_{incr}$ shows less improvement, up to 0.92 ± 0.03, and requires two more sequences (8 and 9) to reach a level comparable to the others. After 9 sequences, VSkNN exhibits the better accumulation performance, (0.99 ± 0.003) closely followed by $AMCS_{CD}$, $AMCS_w$ and $AMCS_{LC}$ (0.97 ± 0.01). $AMCS_{incr}$ exhibits the lowest performance, at 0.95 ± 0.03.

As with the transactional-level results, the proposed $AMCS_{CD}$ is capable of exhibiting similar accumulation performance than VSkNN and $AMCS_{LC}$ variant, but with a significantly lower level of complexity, while outperforming the $AMCS_{incr}$ classifier, which requires more data to accommodate to significantly different concepts.

Focus on individuals 21 and 110: The accumulation performances of the five systems for individual 21 and 110 is presented in Fig. 3.13, and reveals the same observations. With individual 21, which data exhibit less abrupt changes (because only half of the ROIs from Lz_1, Lz_2 and Lz_3 have a profile orientation) all systems perform comparably, as confirmed by a *Kruskall-Wallis* test (with a *p-value* of 0.1). However, for individual 110, the presentation of

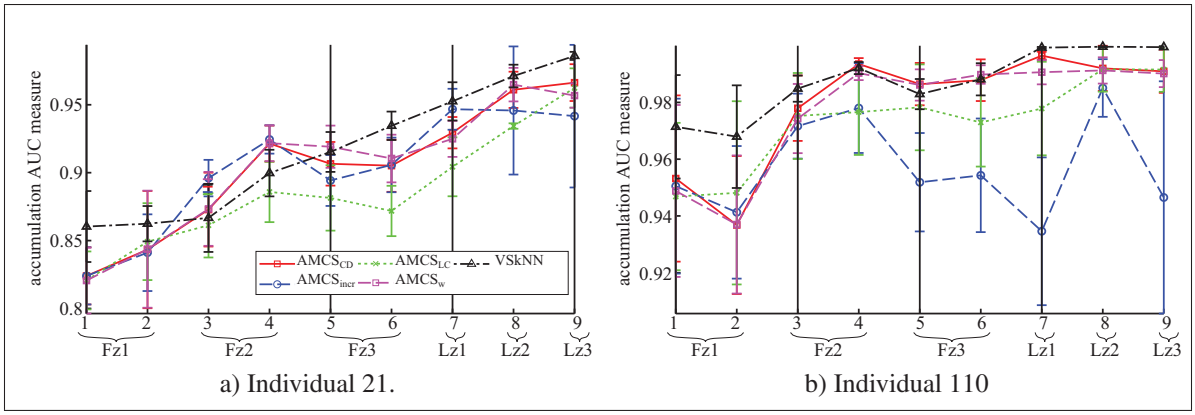


Figure 3.13 Accumulation AUC performance after learning the 9 update sequences.

update sequence 5 (Fz_3 session) decreases $AMCS_{incr}$ accumulation performance from 0.98 ± 0.02 to 0.95 ± 0.02 while the $AMCS_{CD}$ performance remains more stable around 0.99. The significance of this decrease is also confirmed by the *Kruskall-Wallis* test, which confirms that those two system performances are significantly different after the 5th sequence. Similar behaviour can be observed after the presentation of sequence 7 (session Lz_1).

As with the transactional-level analysis, time analysis of the full system reveals the benefits of the proposed change detection strategy. The proposed $AMSC_{CD}$ and $AMCS_w$ are less negatively affected by the introduction of update sequences that incorporate significant concept changes than $AMCS_{incr}$. Yet they achieved comparable performance to VSkNN and $AMCS_{LC}$ with a significantly reduced computational complexity.

3.7 Conclusion

In this paper, a new adaptive multi-classifier system is proposed for video-to-video face recognition in changing environments, as found in person re-identification applications. This modular system is comprised of a classifier ensemble per individual that allows to adapt the facial model of target individuals in response to new reference videos, through either incremental learning or ensemble generation. When a new video trajectory is provided by the operator, a change detection mechanism is used to compromise between plasticity and stability. If the new

data incorporates an abrupt pattern of change w.r.t. previously-learned knowledge (representative of a new concept), a new classifier is trained on the data and combined to an ensemble. Otherwise, previously-trained classifiers are incrementally updated. During operations, faces of each different individual are tracked and grouped over time, allowing to accumulate positive predictions for robust spatio-temporal recognition.

A particular implementation of this framework has been proposed for validation. It consists of an ensemble of 2-class Probabilistic Fuzzy-ARTMAP classifiers for each enrolled individual, where each ensemble is generated and evolved using an incremental training strategy based on a dynamic Particle Swarm Optimization, and the Hellinger Drift Detection Method to detect concept changes. Simulation results indicate that the proposed $AMCS_{CD}$ is able to maintain a high level of performance when significantly different reference videos are learned for an individual. It exhibits higher classification performance than a probabilistic kNN based system adapted to video-to-video FR, as well as a reference open-set TCM-kNN system, with a significantly lower complexity. The scalable architecture employs the change detection mechanism to mitigate the effects of knowledge corruption while bounding its computational complexity.

A key assumption of the adaptive multi-classifier system proposed in this paper is that each trajectory only contains ROI patterns that have been sampled from one concept. In future work, this framework should be extended in order to detect possible sub-concepts in the same trajectory (i.e. changes in facial pose and expression), using for example some windowing strategy. In addition, the particular implementation used for validation has been tested on a large-scaled data set where reference videos have a limited length. Performance should be assessed on other data sets that are representative of person-re-identification (or search and retrieval) applications. Finally, a practical implementation of this framework would require a strategy to purge irrelevant concepts and validation data over time, and bound the system's memory consumption.

CHAPTER 4

DYNAMIC MULTI-CONCEPT ENSEMBLES FOR VIDEO-TO-VIDEO FACE RECOGNITION

Christophe Pagano[1], Eric Granger[1], Robert Sabourin[1], Gian Luca Marcialis[2], Fabio Roli[2]

[1] Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle, École de Technologie Supérieure, Université du Québec, Montréal, Canada

[2] Patter Recognition and Applications Group, Department of Electrical and Electronic Engineering, University of Cagliary, Italy

Article submitted to « Information Sciences » by Elsevier, in 2015.

Abstract

Face Recognition (FR) remains a challenging problem in video surveillance. Facial models of target individuals are typically designed with limited numbers of reference stills or videos captured for an enrollment process. Furthermore, variations in capture conditions contribute to growing divergence between these models and faces captured during operations. Adaptive systems have been proposed for the update of facial models with new facial trajectories that may have been captured under different conditions, and representative of different concepts. Although these systems seek to maintain up-to-date facial models, incremental updating on various concepts may corrupt knowledge. Furthermore, only a subset of this knowledge is typically relevant to classify a given facial capture, and knowledge about completely different concepts may degrade system performance. This paper presents a new framework for adaptive ensembles called Dynamic Multi-Concept Ensemble (DMCE) that is specialized for video-to-video FR. A DMCE is formed with a pool of incremental learning classifiers that is dedicated to an individual enrolled to the system, where each classifier represents different concepts detected in reference trajectories. During enrollment and update phases, multi-modal concept densities are gradually estimated through on-line clustering of reference facial trajectories. Given an input video stream, these densities allow to evaluate the competence of each classifier for faces

captured during operations. The ensemble fusion function is thereby adapted to each facial capture by dynamically weighting classifiers according their relevance for capture conditions. For proof-of-concept, the performance of a particular implementation of DMCE is assessed using videos from the Faces in Action and Chokepoint datasets. Results indicate that the proposed approach provides a higher level of accuracy than reference systems for video-to-video FR and for dynamic selection, while significantly reducing time and memory complexity.

4.1 Introduction

Face recognition (FR) has become a valuable function for several video surveillance (VS) applications, most notably for the detection of individuals of interest over networked video cameras. Some common applications are *watch-list screening* (Bashbaghi *et al.*, 2014) (using still-to-video FR), and *person re-identification* (Pagano *et al.*, 2014) (using video-to-video FR). While each application has its constraints, they all rely on the design of robust face models for matching. A face model of a target individual can be for example a set of template extracted from one or more reference images (galleries of a template matcher), a manifold or statistical model estimated by training with reference images (parameters of a neural network or a statistical classifier), a dictionary learned from reference images (for a sparse representation classifier), a face manifold, etc.

This paper focuses on the design of accurate and robust face classification systems for video-to-video FR in changing surveillance environments, as found in many person re-identification applications. Given faces and sometimes soft biometrics captured in video, these systems provide decision support to operators by recognizing individuals appearing over several live or archived video feeds. To design a face model, Regions of Interest (ROIs) are captured for a same person in reference videos by an operator to enroll individuals of interest appearing on a camera viewpoint. Then, during operations, ROIs captured in live or pre-recorded video streams are matched against face models of target individuals previously enrolled into the system. In this context, the performance of state-of-the-art commercial and academic systems is limited by the difficulty in capturing high quality ROIs from video captured under semi-

controlled (e.g. at inspection lanes, portals and other security checkpoints) and uncontrolled (e.g. in cluttered free-flow scenes at airports or casinos) conditions. Performance is degraded by variations in facial appearance due to changes in pose, scale, orientation, expression, illumination, blur, occlusion and aging.

In VS, many state-of-the-art FR systems process information in terms of face streams or trajectories[1] formed by a person tracker (using faces or soft biometrics). Fusion of system predictions over a trajectory can lead to robust spatio-temporal recognition (Zhou *et al.*, 2006; Barry and Granger, 2007; Matta and Dugelay, 2009). The various conditions under which a face has been captured with video cameras may be represented in a face model for classification as different concepts, i.e. the intra-class data distribution in the input feature space. Given a new reference face trajectory extracted from a video sequence, new concepts can emerge because of variations in camera capture conditions (lighting, occlusion, scale, sharpness, resolution, etc.) and in individual behavior (motion blur facial pose, facial expression, etc.).

An extensive collection of reference ROIs representing all possible capture conditions and camera viewpoint is rarely available for the initial design of facial models in VS applications. While this limits system performance when matching against faces captured under unknown conditions (concepts), new reference trajectories may become available over time for a target individual. This can be used to update and improve the robustness of facial models to intra-class variability. In fact, facial trajectories that are representative of previously-unknown concepts may become available after initial enrollment through some re-enrollment process, self-update, or other sources. Video-to-video FR systems should efficiently adapt face models to integrate these new trajectories without corrupting previously-learned knowledge of the system (other concepts represented in face models) to remain responsive to all possible conditions.

The update of face models over time with new reference trajectories falls within the area of adaptive pattern recognition. Adaptation can either be supervised or semi-supervised, depending on whether reference trajectories are labeled manually by an analyst or automatically

[1] A trajectory is defined as a set of ROIs corresponding to a same high quality track of an individual across consecutive video frames.

according to some confidence function. Moreover, several methods have been proposed for adaptation using either a single incremental learning classifier, or adaptive ensembles of classifiers (Kuncheva, 2004b). In order to preserve previously-acquired knowledge, and yet remain responsive to new information, numerous promising ensemble-based methods have been proposed for adaptation in dynamically changing environments with concept drift (Ortíz Díaz *et al.*, 2015; Polikar *et al.*, 2001; Ramamurthy and Bhatnagar, 2007), some of which are specialized FR systems for VS (Pagano *et al.*, 2014). In this case, adaptation is usually performed by augmenting the pool generated to construct ensembles, either by training new classifiers on newly available data, updating existing classifiers and/or the ensemble fusion function. While the addition of new classifiers can mitigate the effects of knowledge corruption (Pagano *et al.*, 2014; Polikar *et al.*, 2001), a FR system for VS may also benefit from a dynamic adaptation of ensemble fusion rules during operations.

Although temporally related, a trajectory may contain ROIs representative of one or more concepts. It can be typically represented as a multi-modal distribution, modeling these different concepts in the feature space. When operating with a pool of diverse classifiers, each specialized in different concepts, only a subset would therefore be competent for each capture condition. Numerous adaptive ensemble methods propose to update classifier subsets or fusion functions based on the observed concepts (Ortíz Díaz *et al.*, 2015; Ramamurthy and Bhatnagar, 2007), but they are usually designed for concept-drift applications, where the presence of a single concept is considered in the input stream. In addition, these methods only employ a static fusion function that is periodically updated based on the ensemble performance over a recent window of samples (e.g. for a weighted majority vote (Ortíz Díaz *et al.*, 2015)).

Dynamic adaptation (per input ROI) of the ensemble fusion functions may be achieved during operations using dynamic ensemble selection methods. Several dynamic selection methods have been proposed in the literature that evaluate a dynamic region using validation data to assess classifier competence (Britto *et al.*, 2014). For each input ROI captured during operations, a neighborhood may be evaluated within a representative validation data set stored in the system. Then, this neighborhood may be used to evaluate classifier competence w.r.t. to

the inputROI, for example by measuring their classification performance on this labeled data (Woods *et al.*, 1996).

While these methods can improve system accuracy by preventing unrelated classifiers from affecting output predictions, they may interfere with the ability to perform real-time recognition in VS environments. For each input ROI extracted from an operational trajectory, dynamic selection relies on a costly neighborhood evaluation. Time and memory complexity depends on the size of validation data stored in memory, and the size of this validation set would grow over time to represent new concepts. In addition, these methods are limited by the level of intra-class variability represented in facial models (classifier parameters) and validation data. Given a probe ROIs from an unknown concept, incorrect classifier competence estimation is likely to occur because to estimated neighborhood are comprised of data from unrelated concepts, leading to incorrect classifier prediction.

This paper presents a new framework for adaptive ensembles specialized in video-to-video FR, called Dynamic Multi-Concept Ensemble (DMCE). It allows to update facial models with new reference facial trajectories made available after initial enrollment, and to dynamically adapt ensemble fusion functions to changing operating conditions. DMCE has evolved from the framework presented in (Pagano *et al.*, 2014), that relies on a change detection mechanism to guide the adaptation of a pool of incremental learning classifiers when new data become available. A DMCE is designed for each target individual enrolled to the system using a pool of incremental learning classifiers, where each one represents different concepts detected in reference trajectories, and performs adaptation at three different levels: 1) ensemble structure, 2) classifier parameters, and 3) fusion function.

During enrollment and update phases, multi-modal concept densities are gradually estimated through on-line clustering of reference trajectories to represent the observation conditions represented by the classifiers. These allow to perform a dynamic adaptation of ensemble fusion functions for each input ROI during operations. Given a ROI extracted from an operational trajectory, the competence of each classifier is estimated from the ROI's degree of belong-

ing to concept regions, which is then used to weight its fusion function. This method for competence estimation has lower computational complexity than dynamic selection methods involving neighborhood estimation in validation datasets, as well as a more robust competence estimation for input ROIs from unknown concepts, free from classifier-induced biases. Finally, ensemble output responses are accumulated along trajectories formed by a person or face tracker, for robust spatio-temporal prediction.

For validation, a particular implementation of DMCE is proposed. On-line Fuzzy C-Means (Bezdek *et al.*, 1984) is used to cluster facial ROIs, and uncover multi-modal concept models. Individual-specific ensembles of 2-class ARTMAP classifiers (Carpenter *et al.*, 1992) are trained with an incremental learning strategy based on Dynamic Particle Swarm Optimization (Connolly *et al.*, 2012). During operations, a dynamic weighted average fusion rule exploits the Fuzzy C-Means degree of belonging to the closest cluster of each concept to dynamically weight classifiers. The accuracy and resource requirements of this system is assessed using facial trajectories extracted from videos of the Face in Action (Goh *et al.*, 2005) and the Chokepoint (Wong *et al.*, 2011) databases. They are representative of real-world VS applications, and exhibit both gradual and abrupt changes. The performance of the video-to-video FR system implemented with DMCE is compared to the same implemented using three different fusion rules: (1) average score-level fusion, (2) dynamic-selection DS-LA OLA (Woods *et al.*, 1996), and (3), dynamic score-level weighted average using DS-LA OLA accuracy as weights. For references, it is also compared to dynamic adaptations of a probabilistic kNN (Holmes and Adams, 2002), TCM-kNN (Li and Wechsler, 2005), and the adaptive sparse coding FR system (Mery and Bowyer, 2014).

The next section of this paper presents an overview of the literature on FR in VS, followed by adaptive ensemble strategies in Section 3. In Section 4, the new DMCE framework is presented, along with the specific implementation considered for validation. In Section 5, the experimental methodology (video data, protocol and performance measures) is described. Simulation results are then presented and discussed in Section 6.

4.2 Face Recognition in Video Surveillance

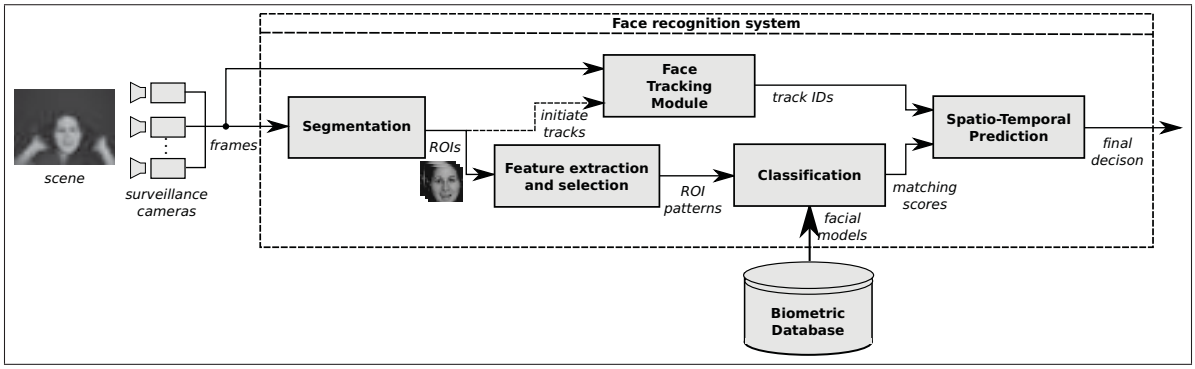


Figure 4.1 General video-to-video face recognition system.

This paper focuses on the design of accurate video-to-video FR systems, embedded as decision support tools for intelligent VS. Individuals of interest are enrolled to the system by an analyst, and their facial models may be refined over time as new reference face trajectories become available. During operations, facial captures from streams provided by a network of digital cameras are matched against these facial models, to alert the analyst to the possible presence of individuals of interest.

Figure 4.1 presents a general system for video-to-video FR. During operations, each camera captures streams of 2D images from a particular viewpoint, which are first processed by a segmentation module to isolate ROIs corresponding to the actual facial regions. Discriminant features are then extracted to generate *ROI patterns* $\mathbf{q}$. These are then matched against the facial model of each individual i stored into the biometric database by the classification module, to produce matching scores $s_i(\mathbf{q})$. In parallel, tracking features $\mathbf{b}$ are extracted to follow the position of ROIs over several frames, regrouping them per track (or trajectory). Tracking information are finally combined with matching scores, to provide robust identity prediction. During enrollment, facial models of each individual i are design a priori using one or more reference ROI patterns $\mathbf{a}^i$, extracted from a video sequence or trajectory. For example, when classification is performed using neural networks (e.g multi-layer perceptrons (Riedmiller, 1994) and ARTMAP neural networks (Carpenter *et al.*, 1991)) or statistical classifiers (e.g. naïve Bayes

classification (Duda and Hart, 1973)), facial models consist of parameters estimated during their training with reference ROI patterns.

Numerous techniques have been proposed for video-to-video FR, combining still FR techniques with spatial and temporal information (Zhou *et al.*, 2006; Barry and Granger, 2007; Matta and Dugelay, 2009). For example, classifier scores may be accumulated over trajectories of correlated ROIs, to increase the reliability of final predictions and reduce ambiguity. However, dedicated systems for VS are not numerous (Pagano *et al.*, 2014). FR in VS is considered as an *open-set* problem, where it is assumed that a majority of faces observed during operations do not belong to individuals of interest. Some specialized architectures have been presented to address this specificity, such as the open-set TCM-kNN, a global multi-class classifier employed with a specialized rejection option for unknown individuals (Li and Wechsler, 2005). In addition, further specialization have been proposed with modular systems designed with individual-specific detectors (one or two-class classifiers). Such class-modular architecture have been shown to outperform global classifiers in applications where the design data is limited w.r.t. the complexity of underlying class distributions and to the number of features and classes (Oh and Suen, 2002; Tax and Duin, 2008).

In addition to increasing classification performance in complex and ill-defined recognition environment, class-modular architectures exhibit multiple advantages for FR in VS applications. They allow to specialize feature subset and decision threshold for each individual, in addition to provide a higher flexibility when adding, updating or removing individuals of interest.

4.3 Adaptive Ensemble Strategies for Video to Video Face Recognition

This paper focuses on the adaptation of facial models given new blocks of reference data (ROI patterns in reference trajectories) becoming available over time. Two families of adaptive methods have been proposed in the literature, either involving the incremental update of a single classifier, or in an ensemble of classifiers (EoCs). On one hand, incremental learning classifiers (such as the ARTMAP (Carpenter *et al.*, 1992) and Growing Self-Organizing (Fritzke,

1996) families of neural networks) are designed to adapt their parameters in response to a new block of data. On the other hand, EoC techniques adapt the ensemble structure (adding/removing classifiers to a pool base), and the selection of classifiers and/or fusion function (Kuncheva, 2004b). The parameters of the classifiers in an ensemble can also be adapted, when ensembles of incremental learning classifiers are considered (Pagano *et al.*, 2014).

As highlighted by the *plasticity-stability* dilemma (Grossberg, 1988), an incremental classifier should remain stable w.r.t. previously-learned concepts, yet allow for adaptation w.r.t. relevant new concepts that emerge in new reference data. While updating a single classifier can translate to low system complexity, it has been observed that the incremental learning of significantly different reference data can corrupt the previously acquired knowledge (Connolly *et al.*, 2012; Pagano *et al.*, 2014; Polikar *et al.*, 2001). This can be detrimental to FR performance in VS environments where different concepts are learned over time, as reference data become available. To address this limitation, adaptive EoC strategies have been successfully applied to FR in VS applications to combine diversified classifiers into an ensemble to improve the system's overall performance and plasticity to new reference data (Pagano *et al.*, 2014).

4.3.1 Generation and Update of Base Classifiers

Numerous techniques have been proposed in literature to adapt classifier ensembles to streams of data with changing underlying distributions. Following the definition of Gama et al. (Gama *et al.*, 2004) and Ditzler et al. (Ditzler and Polikar, 2011), these methods can be differentiated by the way they handle concept drift, either using *passive* or *active* approaches.

Passive methods are designed to continuously adapt to new data without monitoring possible concept drifts, that are handled through automatic adaptation mechanisms. For example, when a new batch of data become available, Boosting methods from the Learn++ family (Muhlbaier and Polikar, 2007; Muhlbaier *et al.*, 2009; Polikar *et al.*, 2001) propose to generate one or several new classifiers, and combine them with previous ones through weighted majority voting.

In contrast *active* methods monitor data streams to detect concept drifts, in which case specific adaptation mechanisms are triggered. For example, in the Just-in-Time classification algorithm for recurring concepts (Alippi *et al.*, 2013), a density-based change detection is used to regroup reference samples per detected concept, and update classifiers using this knowledge when the observed data drift toward a known concept. Similarly, with the Diversity for Dealing with Drifts algorithm (Minku and Yao, 2012), two ensembles with different diversity levels are maintained over time. When a significant discrepancy is detected though the monitoring of the system's error rate during operations, the high diversity ensemble is used to assimilate new data and converge to a low diversity ensemble, and a new high diversity one is generated through bagging. Other methods rely on concept change detection to decide whether to train a new classifier on recent data, or leave the ensemble unchanged (Ramamurthy and Bhatnagar, 2007; Ortíz Díaz *et al.*, 2015). A new classifier is added only if a new concept is detected in the observed data, which limits unnecessary system growth with redundant information. Following the same rationale, Pagano et al. (Pagano *et al.*, 2014) proposed an active EoC approach for FR in VS, with a dedicated ensemble of 2-class incremental classifiers for each enrolled individuals. Changes are detected in the reference data using the Hellinger drift detection method (Ditzler and Polikar, 2011), to only update the ensembles with a new classifier when an abrupt change is detected. In addition, as the ensembles are comprised of incremental Probabilistic Fuzzy-ARTMAP classifiers (Lim and Harrison, 1995), they can be updated when gradual drifts are detected.

In the VS environment considered in this paper, facial models must be updated to integrate knowledge about new concepts and yet preserve previously-observed ones, as they may still be relevant in future operations. In (Pagano *et al.*, 2014), this compromise is addressed through an *active* strategy, mixing ensemble and incremental learning techniques. However, while this method enables to maintain a diverse ensemble with classifiers specialized on the different concepts observed in reference data, a dynamic adaptation should also be considered for operations. Depending on the nature of any ROI pattern extracted from operational streams (i.e. the concept it represents), only a fraction of the classifiers is relevant, and classification perfor-

mance may be increased by preventing unrelated classifiers to affect the final decision. Such adaptation occur on the decision level of an ensemble, either at the selection and/or the fusion stage.

4.3.2 Classifier Selection and Update of Fusion Rule

In addition to updating the pool of base classifiers, adaptive ensembles techniques for concept drift also incorporate strategies to adapt selection and fusion functions. For example, with *horse racing* ensemble algorithms (Blum, 1997; Zhu *et al.*, 2004), a static ensemble of L classifiers is associated with weights that are updated over time depending on their performance over past data. These weights can then be used to perform fusion through weighted majority, or to perform selection by using the prediction of the classifier with the highest weight as the ensemble decision (Hedge β method). Another example is the *Winnow* algorithm (Littlestone, 1988), that only updates a classifier weight when it gives a correct prediction despite the ensemble decision being wrong (promotion step).

Other methods combine strategies to update the pool of base classifiers in addition to the fusion rule. For example, the Learn++.NSE variant (Muhlbaier and Polikar, 2007) relies on weighted majority voting for fusion, and proposes to keep track of the performance of each classifier of the ensemble w.r.t. past batches of operational data. These measurements are used as voting weights, and are updated to integrate new batches of data, giving more weight to recent measurements. When a recurring concept is re-encountered, historical measures enable to detect the presence of a known concept, and increase the weights of related classifiers. In the same way, the Fast Adapting Ensemble method (Ortíz Díaz *et al.*, 2015) implements heuristics to either activate or deactivate classifiers depending on the detected concept, as only activated classifiers participate in the final decision. When the presence of a previously encountered concept is detected in the operational data, classifiers associated to this concept re-activated, and their weights are adjusted.

However, the methods described above only apply a static selection or fusion rule (Britto *et al.*, 2014). The fusion parameters are updated a posteriori after an observation over a window of past data, and remain the same for every ROI pattern until the next update. In addition, they are designed for concept drift applications, where the stream of operational data is monitored for possible drifts toward new concepts. Concept evaluation, and thus adaptation, only assumes the presence of a single concept in the input stream. This assumption is not always valid in FR in VS applications. In a trajectory of an individual's face, multiple concepts can be observed, for example corresponding to changes in facial pose w.r.t. the camera during the same sequence.

A dynamic adaptation of the fusion rule has been proposed by Jacobs et al. with the Mixture of Experts system (Jacobs *et al.*, 1991). It is comprised of an ensemble of neural network classifiers, as well as an additional gating network. For each input, the gating network computes the probabilities that each neural network of the ensemble is the most competent to classify it. These probabilities are then used to compute the fusion function, as the weighted average of the network outputs. Although providing dynamic weighting, the architecture of this method remain static, as the gating network has to be re-initiated from the start with previous and new reference data to remain relevant. It may also require the storage of previous data, for example to adapt its structure to the addition of a new classifier in the ensemble. Dynamic selection methods have also been proposed in the literature to provide a dynamic competence estimation of the most relevant base classifiers per input pattern (Britto *et al.*, 2014). For each input to classify, these methods involve the computation of a region of competence Ψ defined as its k nearest neighbours in a set of validation data of known labels. Numerous methods have been proposed to compute classifier competence from the region Ψ. For example, in (Woods *et al.*, 1996), the accuracy of each classifier is computed as the percentage of correctly classified samples from Ψ, and the classifier with the highest accuracy is selected for classification. Another example is the *DS-KNN* method (Santana *et al.*, 2006), that proposes to determine an optimal ensemble subset instead of a single best classifier, considering both accuracy and ensemble diversity measures. The N' most accurate classifiers in Ψ are first selected to generate an inter-

mediate ensemble. Then, only the N'' most diverse classifiers of this ensemble are selected for classification, using double-fault diversity measures.

However, the performance of dynamic selection methods depends heavily on the storage of a representative set of validation data to estimate classifier competence. To increase the robustness of this representation of intra-class variability, the validation set is likely to grow over time as new concepts are observed in reference trajectories. In addition, the estimation of competence regions Ψ involve a computationally intensive nearest neighbor estimation for each input capture, where computational complexity grows with the size of the validation set. This can which significantly reduce system response time, and its ability to rapidly detect individuals. Finally, although the dynamic computation of competence regions enables to benefit from the most relevant information, these methods remain sensitive to the presence of unknown concepts in the operational streams. When presented with ROIs captured under conditions not represented in facial models nor validation data used to estimate competence, incorrect competence prediction is likely to occur, either because of ill-defined competence regions Ψ (comprised of data from unrelated concepts), or poor classifier performance. In FR in VS, the dynamic adaptation of ensemble fusion function should not interfere with the ability to perform rapid recognition, nor corrupt system performance when unknown concepts are observed during operations.

4.4 Dynamic Multi-Concept Ensembles of Classifiers

In this paper, a new adaptive ensemble framework, called Dynamic Multi-Concept Ensemble (DMCE), is proposed for video-to-video FR (see Fig. 4.2). The DMCE framework is designed to update facial models with newly available reference trajectories over time, and dynamically adapt its behavior during operations based on changing face capture conditions in video streams.

A DMCE is comprised of a pool of incremental classifiers $P^i = \{IC^i_1, IC^i_2, ..., IC^i_{O^i}\}$ per enrolled individual i, where O^i is the number of changes detected in reference trajectories. It relies on a

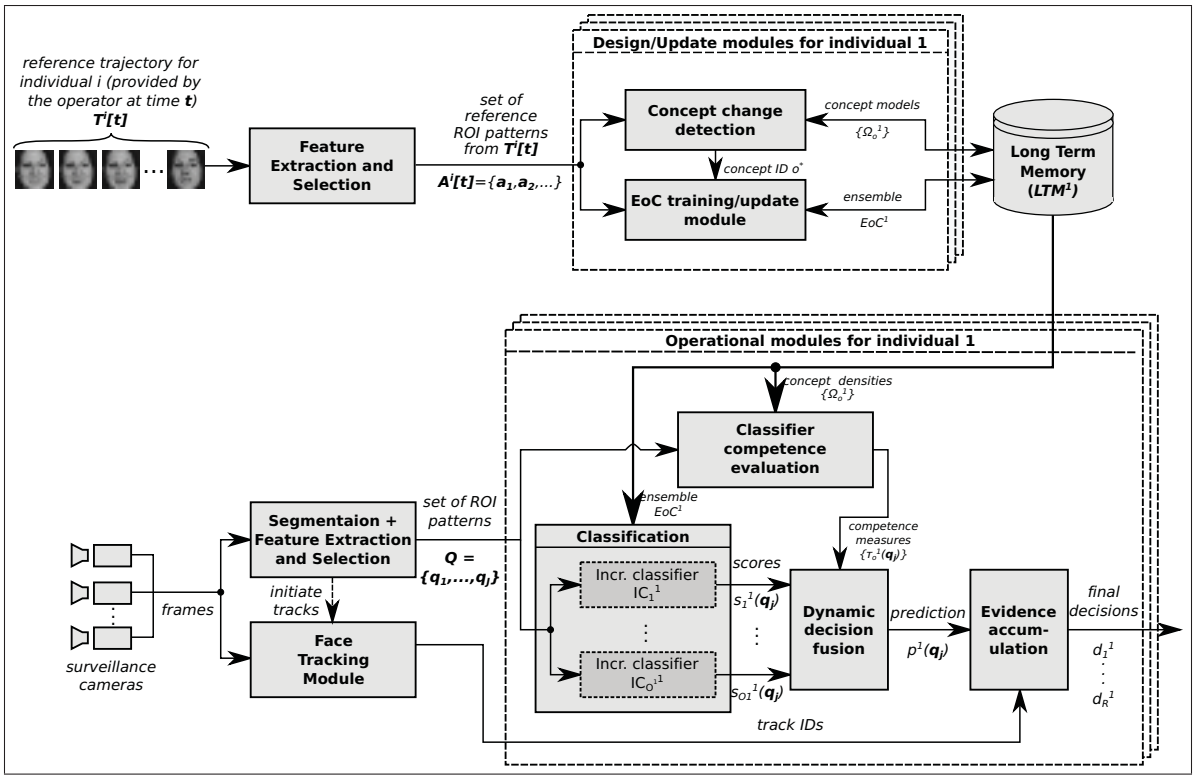


Figure 4.2 Architecture of a system for video-to-video FR based on the proposed DMCE framework.

change detection mechanism to guide the updating strategy for each reference trajectory $T^i[t]$ available at time t. To account for intra-class variations in trajectories, a concept detection module estimates concept densities Ω_o^i ($o = 1,...O^i$) representing overlapping regions of competence of the ensemble classifiers. During operation, these densities are used to evaluate the competence of each classifier for a given input ROI pattern $\mathbf{q}$, such that the ensemble fusion functions are dynamically adapted.

4.4.1 General Framework

4.4.1.1 Design and Update Architecture

For each target individual i enrolled to the system, the design and update architecture is composed of modules for:

- **Long Term Memory**, to store the classifier pools $P^i = \{IC_1^i, IC_2^i, ..., IC_{O^i}^i\}$ and the concept models Ω_o^i $(o = 1, ...O^i)$ for future updates and system operations.
- **Concept change detection**, to detect abrupt changes between reference trajectories $T^i[t]$ and concept densities Ω_o^i $(o = 1, ...O^i)$.
- **EoC training/update**, to update or train new incremental classifiers IC_{o*}^i with new reference trajectories.

DMCE framework can be implemented with multiple types of classifier and change detection method, as long as classifiers can perform incremental learning (to be updated with new trajectories from similar concepts), and concepts densities are modeled on-line as multi-modal distributions in the feature space (to represent intra-class variability in trajectories).

Algorithm 4.1: Design and update procedure for individual i.

Input: *Reference trajectory for individual i* $T^i[t]$*, provided at time* t;
Output: *Updated Long Term Memory* LTM^i;
- Compute $\mathbf{A}^i[t]$, the set of reference ROI patterns obtained after feature extraction and selection of ROIs from $T^i[t]$;
- Compute $\mathscr{A}^i[t]$, the concept density of $\mathbf{A}^i[t]$;
- Perform change detection to determine the index o^* of the closest concept density to $\mathscr{A}^i[t]$;

if *concept change detected* **then**
    - Create a new concept density $\Omega_{O^i+1}^i \leftarrow \mathscr{A}^i[t]$;
    - Update $O^i \leftarrow O^i + 1$;
    - Train a new classifier $IC_{O^i}^i$ with $\mathbf{A}^i[t]$;
    - Update the pool $P^i = P^i \cup IC_{O^i}^i$;
else
    `//Gradual change detected`
    - Update the concept densities $\Omega_{o^*}^i = \Omega_{o^*}^i \cup \mathscr{A}^i[t]$;
    - Update $IC_{o^*}^i$ with $\mathbf{A}^i[t]$;
end
- Store concept densities Ω_o^i $(o = 1, ..., O^i)$ and updated pool P^i into LTM^i;

A general enrollment and update procedure is presented in Alg. 4.1. Once the set of ROI patterns $\mathbf{A}^i[t]$ has been extracted from $T^i[t]$, their corresponding concept density $\mathscr{A}^i[t]$ is estimated. Change detection is then performed between $\mathscr{A}^i[t]$ and the stored concept densities to select the index o^* of the closest density $\Omega^i_{o^*}$. If a gradual change is detected, $\mathbf{A}^i[t]$ is combined with selected non-target ROI patterns to update the corresponding classifier $IC^i_{o^*}$. On the other hand, if an abrupt change is detect, a new density $\Omega^i_{O^i+1}$ is stored, and a new classifier $IC^i_{O^i+1}$ is trained with $\mathbf{A}^i[t]$ combined with selected non-target ROI patterns, and added to P^i. Finally, the updated set of concept densities Ω^i_o ($o = 1,...,O^i$) and the new pool P^i are stored into LTM^i for future update and operations.

4.4.1.2 Operational Architecture

For each individual i, the operational architecture is composed by modules for:

- **Classification**, using the pool $P^i = \{IC^i_1, IC^i_2, ..., IC^i_{O^i}\}$ to compute matching scores $s^i_o(\mathbf{q})$ ($o = 1,...,O^i$).
- **Classifier competence evaluation**, to measure the competence of each classifier $\tau^i_o(\mathbf{q})$ ($o = 1,...O^i$) w.r.t. each input pattern $\mathbf{q}$.
- **Dynamic decision fusion**, to compute predictions $p^i(\mathbf{q})$, combining classifier scores $s^i_o(\mathbf{q})$ and competence measures $\tau^i_o(\mathbf{q})$ ($o = 1,...O^i$).
- **Tracking**, to follow the different individuals across camera view points, and regroup ROIs of a person intro a trajectory.
- **Evidence accumulation**, to accumulate predictions of each pool according to trajectories. This general track-and-classify strategy has been shown to provide a high level of performance in video-based FR Matta and Dugelay (2009).

A general operational procedure is presented in Alg. 4.2, for each individual of interest i, considering a track T captured by a surveillance camera. For each frame, an ROI is first extracted,

Algorithm 4.2: Operational procedure for individual i over a trajectory T.

Input: *Stream of input frames from T, pool P^i and Long Term Memory LTM^i*;
Output: *Final decision d^i for track T*;
for *each frame* **do**
    - Detect ROI and compute pattern $\mathbf{q}$;
    for *each concept density* $o = 1, ..., O^i$ **do**
        - Compute $\tau_o^i(\mathbf{q})$ the competence of classifier IC_o^i w.r.t. $\mathbf{q}$;
    end
    for *each classifier* IC_o^i, $o = 1, ..., O^i$ **do**
        - Compute $s_o^i(\mathbf{q})$, the classification score for individual i;
    end
    - Compute the prediction $p^i(\mathbf{q})$ through dynamic weighting or selection of classifiers;
    - Accumulate $p^i(\mathbf{q})$ into the final decision d^i for track T;
end

and the corresponding pattern $\mathbf{q}$ computed. Competence measures $\tau_o^i(\mathbf{q})$ and classification scores $s_o^i(\mathbf{q}_k)$ ($o = 1, ..., O^i$) related to input $\mathbf{q}$ are then computed for every classifier in P^i. These are combined through dynamic weighting or selection to obtain the prediction $p^i(\mathbf{q})$. The predictions for each input ROI in the trajectory T are finally accumulated into the final decision d^i.

4.4.2 Concept Densities and Dynamic Weighting

Variations in capture conditions, camera properties and individuals' behavior tend to generate complex multi-concept facial models in VS applications. While an active ensemble methodology that trains new classifiers when a concept change is detected in reference trajectories may allow to gradually represent this intra-class variability in facial models, an additional level of adaptation may be required during operations, as abrupt changes may also be observed within facial trajectories. For example, when a trajectory captured by a specific camera viewpoint is likely to be comprised of a majority of profile face poses, frontal poses can occasionally be captured when the individual's head moved towards the camera's direction. These facial poses could be related to the same concept as the majority of ROIs captured by a different camera.

Similarly, a small fraction of ROIs belonging to two different reference trajectories which comparison would trigger an abrupt change could also be related to the same concept. The active updating methodology presented in Algorithm 4.1 may thus generate overlaps in the regions of competence of the classifiers.

Accordingly, DMCE proposes to perform concept change detection using multi-modal concept densities. This enables to implement an active ensemble strategy that has been shown to improve system performance (Pagano *et al.*, 2014), as well as provide a more accurate concept representation accounting for possible overlaps in classifiers' regions of competence. While an abrupt concept change would be detected between trajectories comprised of a majority of samples from different concepts, possible overlaps may be represented by common density modes in the feature space. These densities allow to provide DMCE with an additional level of adaptation during operations. A dynamic ensemble fusion rule is proposed for weighting the influence of each classifier according to their competence, estimated as the closeness of each probe ROI to the corresponding concept densities.

For a specific implementation, densities should be estimated with a method that allows for soft associations, to account for overlap and dispersion in regions of competence. In addition, a suitable method would allow to model multi-modal distributions from a limited amount of ROI patterns, as their amount in reference trajectory may be less that the dimensionality of the density feature space. Finally, densities should be updated on-line over time as new reference trajectories become available, without requiring access to previously-captured data.

An illustration is provided in Figure 4.3. The DMCE training methodology applied to individual 201 of the FIA dataset (Goh *et al.*, 2005) detects 5 abrupt changes. In this figure, only two concepts densities are represented, respectively Ω_1^{201} and Ω_4^{201}, with 3 clusters each. Reference patterns used to generate these models have been projected into a 2D space using Sammon mapping, and associated to their respective clusters by color. In addition, the closest ROI pattern to each cluster is presented, to illustrate the relation between clusters and real operating conditions. In this example, a multi-modal concept representation enables to model an over-

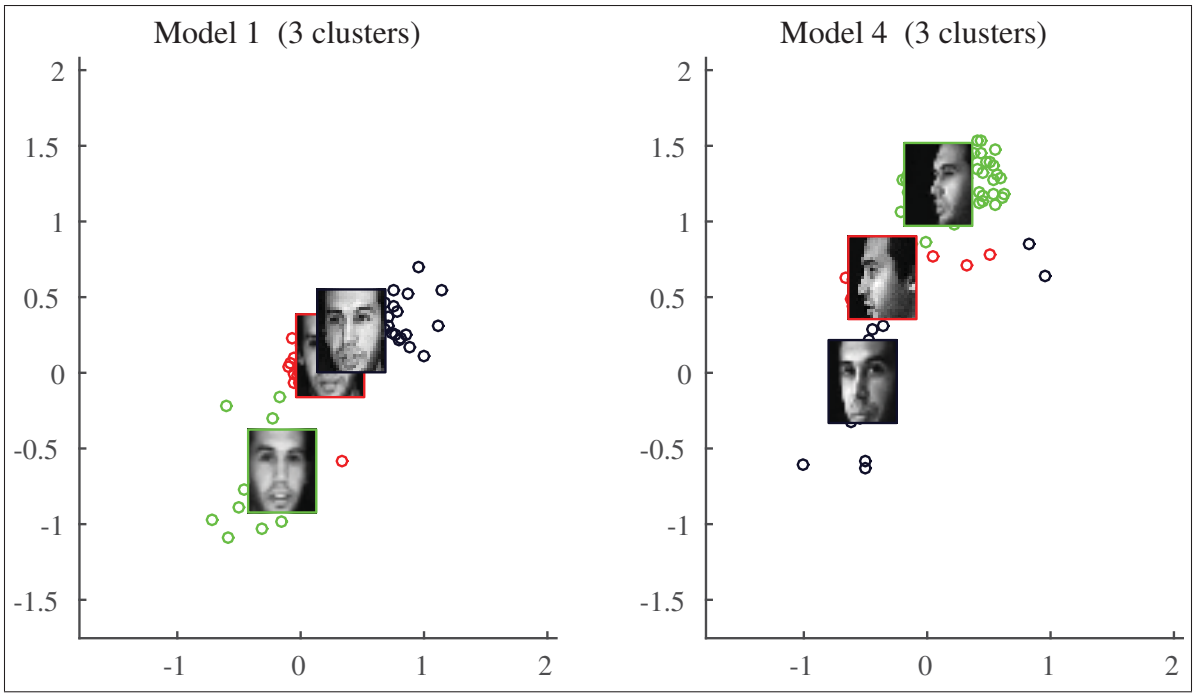


Figure 4.3 2D projection of 2 of the 5 concept densities (1 and 4) generated by the change detection module of DMCE for individual 201 of the FIA dataset Goh *et al.* (2005), following the protocol presented in Section 4.5. For each concept density, reference patterns used for its generation have been projected into a 2D space using Sammon mapping. Each cluster is presented in a different color, along with the ROI associated to the closest pattern to its center.

lapping competence region between classifier IC_1^{201} and IC_4^{201}. While model Ω_4^{201} has been generated with a majority of profile views (2 clusters out of 3), it can be observed that the bottom left cluster is comprised of semi-frontal views, located in an area close to the frontal view clusters of model Ω_1^{201}.

For each input ROI pattern $\mathbf{q}$ captured during operations, this information is then exploited by DMCE to dynamically adapt the ensemble fusion function to $\mathbf{q}$. In DMCE, competence measures $\tau_o^i(\mathbf{q})$ are computed as functions of the distances between $\mathbf{q}$ and the closest cluster of each concept density. Figure 4.3, with an input pattern $\mathbf{q}$ corresponding to a profile view, competence $\tau_4^{201}(\mathbf{q})$ would then necessarily be higher than $\tau_1^{201}(\mathbf{q})$. However, for an input pattern corresponding to a frontal view close to the lower left corner of the Sammon space

of Figure 4.3, the two classifiers would have closer competence measures, and thus similar influence in the final decision.

4.4.3 Specific Implementation

A specific implementation of DMCE is proposed for the experiments of this paper. This section only presents the details related to the main contributions of DMCE: (1) concept density models, (2) change detection procedure, and (3), dynamic competence evaluation and decision fusion. Further implementation details are presented in the next section on methodology.

4.4.3.1 Concept Densities

Each classifier IC_o^i is associated to a concept density $\Omega_o^i = \{\mu_{o,l}^i, n_{o,l}^i\}$ ($l = 1, ..., L_o^i$), with L_o^i the number of clusters, $\mu_{o,l}^i$ the center of cluster l in the feature space and $n_{o,l}^i$ the number of reference patterns associated to it during training. Clusters may overlap and be dispersed in the feature space, and the ROIs captured during operations may not exactly fit the concept regions detected during training. For these reasons, clustering methods that allow for soft cluster associations to model concept overlap are considered to estimate densities from reference trajectories. In addition, on-line methods are considered to avoid storing previous data and re-start clustering from scratch when a concept density is updated.

Incremental clustering methods using Gaussian Mixtures Models (GMMs) have been proposed to update densities with new batches of data (Song and Wang, 2005). However, in the VS scenario considered in this paper, the limited amount of ROI patterns in reference trajectory v.s. the dimentionality of the feature space would generate ill-defined distributions. For this reason, an incremental approach for clustering with Fuzzy C-means (FCM) (Bezdek *et al.*, 1984) using the incremental methodology of (Song and Wang, 2005) is considered in this paper. FCM has been selected for the low computational and memory complexity of the cluster belonging function. In this method, clustering is first performed on the ROI patterns from the new trajectory, and the resulting density is combined with the old concept density that has to be

updated. More precisely, if a gradual change is detected between $\mathscr{A}^i[t]$ and a concept density Ω^i_{o*}, $\mathscr{A}^i[t]$ is considered to be comprised of a majority of clusters similar to Ω^i_{o*}. In this case, the two densities are merged, and the classifier associated to Ω^i_{o*} is updated using $\mathbf{A}^i[t]$.

Algorithm 4.3: Merging of concept densities $\mathscr{A}^i[t]$ and $\Omega^i_{o^*}$.

Input: *Trajectory density* $\mathscr{A}^i[t]$*, concept density to update* $\Omega^i_{o^*}$;
Output: *Updated concept density* ${\Omega'}^i_{o^*}$;
- Initialize ${\Omega'}^i_{o^*} \leftarrow \emptyset$;
- Compute $\hat{\delta}^i_f$ and $\hat{\sigma}^i_f$, the average and standard deviation of the distances between each center of $\Omega^i_{o^*}$;
- Compute the fusion threshold $\gamma^i_{o^*} = \hat{\delta}^i_f - \hat{\sigma}^i_f$;
for *each center* $\mu^i_{A,l'}[t] \in \mathscr{A}^i[t]$ **do**
    for *each center* $\mu^i_{o^*,l^*} \in \Omega^i_{o^*}$ **do**
        - Compute $\delta^i_f(l,l') = d_{Eucl}(\mu^i_{o^*,l}, \mu^i_{A,l'}[t])$;
    end
    - Find $l^* = argmin\{\delta^i_f(l,l') : l = 1,...,L^i_{o^*}\}$ the index of the closest center $\mu^i_{o^*,l^*}$ from $\Omega^i_{o^*}$ to $\mu^i_{A,l'}[t]$;
    if $\delta^i_f(l^*,l') < \gamma^i_{o^*}$ **then**
        - $\mu^i_{o^*,l^*}$ and $\mu^i_{A,l'}[t]$ are *close*;
    else
        - Update ${\Omega'}^i_{o^*} = {\Omega'}^i_{o^*} \cup \{\mu^i_{A,l'}[t], n^i_{A,l'}[t]\}$;
    end
end
for *each center* $\mu^i_{o^*,l^*} \in \Omega^i_{o^*}$ **do**
    for *each center* $\mu^i_{A,l'}[t] \in \mathscr{A}^i[t]$ **do**
        if $\mu^i_{o^*,l}$ *and* $\mu^i_{A,l'}[t]$ *are close* **then**
            - Update $\mu^i_{o^*,l} = \frac{n^i_{o^*,l} \cdot \mu^i_{o^*,l} + n^i_{A,l'}[t] \cdot \mu^i_{A,l'}[t]}{n^i_{o^*,l} + n^i_{A,l'}[t]}$;
            - Update $n^i_{o^*,l} = n^i_{o^*,l} + n^i_{A,l'}[t]$;
        end
    end
    - Update ${\Omega'}^i_{o^*} = {\Omega'}^i_{o^*} \cup \{\mu^i_{o^*,l}, n^i_{o^*,l}\}$;
end

The density merging process is presented in algorithm 4.3. First of all, the fusion threshold is computed following $\gamma^i_{o^*} = \hat{\delta}^i_f - \hat{\sigma}^i_f$, with $\hat{\delta}^i_f$ and $\hat{\sigma}^i_f$ respectively the average and standard deviation of the distances between each center of $\Omega^i_{o^*}$. This ensures that the updated clustering structure remains similar after fusion. The distances $\delta^i_f(l,l')$ ($l = 1,...,L^i_{o^*}$, $l' = 1,...,L'$) are then computed between each centers of the two densities, to find, for each center $\mu^i_{A,l'}[t] \in \mathscr{A}^i[t]$ the closest center $\mu^i_{o^*,l^*} \in \Omega^i_{o^*}$. If the distance $\delta^i_f(l^*,l')$ is lower than $\gamma^i_{o^*}$, the two clusters are considered *close*. Then, every center $\mu^i_{A,l'}[t] \in \mathscr{A}^i[t]$ not *close* to any center from $\Omega^i_{o^*}$ is added to the final updated density Ω'^i_{o*}. Finally, each center $\mu^i_{o^*,l} \in \Omega'^i_{o*}$ is also added to Ω'^i_{o*}, depending on its nature:

- If $\mu^i_{o^*,l}$ is not *close* to any center: $\mu^i_{o^*,l}$ is added *as-is*.
- If $\mu^i_{o^*,l}$ is *close* to at least one center: $\mu^i_{o^*,l}$ is merged with every *close*, center as a center of mass weighted by the number of associated samples, and added to the model.

4.4.3.2 Change Detection

For each new reference trajectory with ROI patterns $\mathbf{A}^i[t]$, changes are detected between the corresponding concept density $\mathscr{A}^i[t]$ and each stored density Ω^i_o by monitoring the following distance measure:

$$\delta^i_c(o,t) = \frac{1}{L^i_o \,.\, L'} \sum_{l=1}^{L^i_o} \sum_{l'=1}^{L'} d_{Eucl}(\mu^i_{o,l}, \mu^i_{A,l'}[t]) \tag{4.1}$$

with $d_{Eucl}(\mu^i_{o,l}, \mu^i_{A,l'}[t])$ the Euclidean distance between $\mu^i_{o,l}$ and $\mu^i_{A,l'}[t]$. This distance is then compared to a dynamic threshold defined by:

$$\Gamma^i_o = \hat{\delta}^i_c - \hat{\sigma}^i_c \tag{4.2}$$

where $\hat{\delta}^i_c$ and $\hat{\sigma}^i_c$ are respectively the average and standard deviation of past $\delta^i_c(o,t)$ measures. This measure allows to detect if an abrupt change occurred between the concepts represented by $\mathscr{A}^i[t]$ and the other concept densities stored in memory. If $\mathscr{A}^i[t]$ is comprised of a majority

of new clusters (abrupt change), a new classifier is trained with the corresponding data, and a new concept density added to the system as $\Omega^i_{O^i+1} = \mathscr{A}^i[t]$.

4.4.3.3 Dynamic Competence Evaluation and Decision Fusion

In the proposed system, each classifier IC^i_o ($o = 1,...,O^i$) is associated to a concept density Ω^i_o representing the reference data used for its training. For each input pattern $\mathbf{q}$, the closer $\mathbf{q}$ is to Ω^i_o, the more competent IC^i_o is considered for classification. For each concept density, the competence measure is computed from the Fuzzy C-means (Bezdek *et al.*, 1984) degree of belonging to closest cluster, following:

$$\tau^i_o(\mathbf{q}) = w_{l*}(\mathbf{q})^m = \frac{1}{d_{Eucl}(\mathbf{q}, \mu^i_{o,l*})^m} \tag{4.3}$$

with $l* = argmax\{w_l(\mathbf{q})^m : l = 1,...,L^i\}$ the index of the closest cluster of Ω^i_o, and m the *fuzziness* parameter of Fuzzy C-means.

The competence measures $\tau^i_o(\mathbf{q})$ ($o = 1,...,O^i$), as well as the classification scores $s^i_o(\mathbf{q})$ are combined to obtain the final score $S^i(\mathbf{q})$ following:

$$S^i(\mathbf{q}) = \frac{\sum_{o=1}^{O^i} \tau^i_o(\mathbf{q}).\, s^i_o(\mathbf{q})}{\sum_{o'=1}^{O^i} \tau_{o^i}(\mathbf{q})} \tag{4.4}$$

A score-level dynamic weighted average is considered for decision fusion since it provides a reliable dynamic adaptation in the following three cases:

a. when $\mathbf{q}$ is close to a known concept region: the influence of competent classifiers is increased through higher weights.

b. when $\mathbf{q}$ is close to a concept region that's been observed in the reference data of several classifiers: competence overlap is accounted for by providing similar weights for related classifiers.

c. when **q** is sampled from an unknown concept: every classifier contribute to the final decision in a similar way, to rely on the diversity of the full ensemble to classify samples related to previously-unknown concepts.

4.5 Experimental Protocol

4.5.1 The Faces in Action Dataset

4.5.1.1 Dataset presentation

The Carnegie Mellon University Faces In Action (FIA) face database (Goh *et al.*, 2005) is the first dataset considered for experimental validation of DMCE. It is composed of 20-second videos capturing the faces of 221 participants in both indoor and outdoor scenario, each video mimicking a passport checking scenario. Videos have been captured at three different horizontal pose angles (0° and $\pm 72.6^\circ$), each one with two different focal length (4 and 8mm). For the experiments of this paper, all ROIs have been segmented from each frame, using the OpenCV v2.0 implementation of the Viola-Jones algorithm (Viola and Jones, 2004), and the faces have been rotated to align the eyes (to minimize intra-class variations (Gorodnichy, 2005a)). ROIs have been scaled to a common size of 70x70 pixels, which was the smallest detected ROI, before fetaure extraction. The FIA videos have been separated into 6 subsets, according to the different cameras (left, right and frontal face angle, with 2 different focal length, 4 and 8 mm) for each one of the 3 sessions, and for each individual. Only indoors videos for the the frontal angle (0°) and left angle ($\pm 72.6^\circ$) are considered for experiments in this paper.

4.5.1.2 Simulation scenario

The same simulation scenario is considered than in (Pagano *et al.*, 2014). Ten (10) individuals of interests have been selected as target individuals, subject to two experimental constraints: 1) they appear in all 3 sessions, and 2), at least 30 ROIs for every frontal and left videos have been detected by the OpenCV segmentation. The ROIs of the remaining 200 individuals are mixed

into a Universal Model (UM), to provide classifiers with non-target samples. Only 100 of those individuals have been randomly selected for the training UM, to ensure that the scenario contains unknown individuals in testing (i.e.the remaining 100 whose samples have never been presented to the system during training). To avoid bias due to the more numerous ROI samples detected from the frontal sessions, the original FIA frontal sets have been separated into two sub-trajectories, forming a total of 9 reference trajectories for design and update (see Table 4.1). Simulations emulate the actions of a security analyst in a decision support system, that provides the systems with new reference trajectories $T^i[t]$ to update the face models of individuals $i = 1,..,10$ at a discrete time $t = 1,2,...,9$.

Table 4.1 Correspondence between the 9 reference trajectories of the experimental scenario and the original *FIA* video sequences.

Time step t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Reference Trajectory	$T[1]$	$T[2]$	$T[3]$	$T[4]$	$T[5]$	$T[6]$	$T[7]$	$T[8]$	$T[9]$
Corresponding FIA sequence	Frontal cam. session 1		Frontal cam. session 2		Frontal cam. session 3		Left cam. session 1	Left cam. session 2	Left cam. session 3

Reference trajectories are selected from the cameras with 8-mm focal length in order to provide ROIs with better quality for training. ROIs captured during 3 different sessions and 2 different pose angles may be sampled from different concepts, and the transition from sequence 6 to 7 (change of camera angle) represents most abrupt concept change in the reference ROI patterns. Changes observed from one session to another, such as from trajectory 2 to 3, 4 to 5, 7 to 8 and 8 to 9 depends on the individual. As faces are captured over intervals of several months, both gradual and abrupt changes may be detected.

For each time step $t = 1,2,...,9$, the systems are evaluated after adaptation on the same test dataset, emulating a practical security checkpoint station where different individuals arrive one after the other. The test dataset is composed by trajectories from every session and pose angle to simulate face re-identification applications where different concepts may be observed during operations, but where the analyst gradually tags and submits new trajectories to the

system to adapt face models. Every different concept (face capture condition) for which the system can adapt is present in the test data, and thus should be preserved over time. In order to present different facial captures than the ones used for training, only the cameras with 4-mm focal length are considered for testing. While every facial capture is scaled to a same size, the shorter focal length adds additional noise (lower quality ROIs), which simulates a real-life scenario where reference and operational ROI patterns are not necessarily related to the same capture conditions.

4.5.2 The ChokePoint Dataset

4.5.2.1 Dataset presentation

The Chokeoint Dataset (Wong *et al.*, 2011) has been designed for person identification and verification scenarios under real-world video-surveillance conditions. Videos from respectively 25 and 29 individuals have been captured as they walked naturally through two portals comprised of three cameras each, placed at natural choke points. For each portal, 4 capture sessions have been performed, each time recording the individuals entering and leaving the portal. Due to unconstrained observation conditions, variations in illumination, pose, sharpness and partial occlusion can be observed in these captures. The dataset contains a total of 48 video sequences, from which 64,204 face images have been extracted by the dataset authors using manually labelled eye position. These ROIs have been scaled to a common size of 96x96 pixels. For the experiments of this paper, the two groups G1 and G2 proposed by the dataset authors for video-to-video verification are considered. Each group is comprised of 8 video sequences for all available individuals, only selecting sequences with the post frontal pose views. The contents of each group are presented in Table 4.2. In this paper, group G1 is considered for training, and G2 for testing.

Table 4.2 Chokepoint verification protocol, presented in (Wong *et al.*, 2011). Sequences are named according to their capture conditions, with P,S and C respectively standing for *portal*, *session* and *camera*, and *E* and *L* indicating if the subjects are entering or leaving the portals.

G1	P1E_S1_C1	P1E_S2_C2	P2E_S2_C2	P2E_S1_C3
	P1L_S1_C1	P1L_S2_C2	P2L_S2_C2	P2L_S1_C1
G2	P1E_S3_C3	P1E_S4_C1	P2E_S4_C2	P2E_S3_C1
	P1L_S3_C3	P1L_S4_C1	P2L_S4_C2	P2L_S3_C3

4.5.2.2 Simulation scenario

A similar simulation scenario than proposed for the FIA dataset is considered. Individuals of interest have been selected with the same constraints, i.e. present in each training and testing session, with at least 30 ROIs in each training session. 23 individuals fulfill these constraints, and only the first 10 are considered as individuals of interest, leaving the remaining 19 of the dataset as non-targets. For each time step $t = 1, ..., 8$, the compared systems are updated with reference images from trajectory number t of G1. In the same way than with the FIA dataset, performances are evaluated at each time step on all sequences of the G2 group.

4.5.3 Reference systems

The performance of the proposed implementation of DMCE is compared to three variants of the same system, using different fusion rules:

a. *AMCS* system, presented in (Pagano *et al.*, 2014), which can be considered as a implementation of DMCE with score-level average fusion.

b. *DMCE with DS-LA OLA selection*, using the local accuracy (LA) dynamic selection method presented in (Woods *et al.*, 1996). For each input pattern $\mathbf{q}$, competence measures $\tau_o^i(\mathbf{q})$ ($o = 1, ..., O^i$) are computed as overall local accuracy (OLA) of the classifiers within a neighborhood in a validation dataset, as described in Algorithm 4.4. The com-

bined score $S^i(\mathbf{q})$ is determined as the score of classifier $IC^i_{o^*}$, associated to the highest competence measure $\tau^i_{o^*}(\mathbf{q})$.

c. *DMCE with DS-LA OLA weighting*, competence measures computed following Algorithm 4.4 are used as dynamic weights for score-level weighted average fusion.

In addition, 3 reference systems are considered:

a. An adaptive and class-modular version of the open-set TCM-kNN (Li and Wechsler, 2005), previously been applied to video-to-video FR. To adapt its whole architecture, its parameters are also updated at every time step, as well as the value of k (for the kNN) which is validated through (2x5 folds) cross validation. Finally, a final decision threshold Θ^i is validated for each individual of interest using the same methodology than DMCE.

b. VSkNN, a probabilistic class-modular k-NN classifier, adapted to VS. A separate k-NN classifier using Euclidean distance is considered for each individual of interest i, trained using positive reference samples from video sequences of target individual i, and a mixture of negative reference samples from the UM and CM, as with DMCE. A score is then computed through the *probabilistic kNN* approach (Holmes and Adams, 2002): the probability of the presence of the individual i is the proportion, among the k nearest neighbours, of reference samples from the same individual. The value of k is also validated through (2x5 folds) cross validation, along with the final decision threshold Θ^i.

c. A FR system using Adaptive Sparse Representations (ASR) of random patches, presented in (Mery and Bowyer, 2014). Batch learning is considered to adapt to new reference data, and standard parameters are used, except for the number of patches that is reduced to 100 due to memory constraints.

Algorithm 4.4: Competence computation for DMCE DS-LA OLA variants.

Input: *Input pattern* $\boldsymbol{q}$;
Input: *Pool* EoC^i *and Long Term Memory* LTM^i;
Input: *Competence validation dataset dbVal*;
Output: *Competence measures* $\tau_o^i(\boldsymbol{q})$ $(o = 1, ..., O^i)$;
- Estimate the competence region $\Psi \leftarrow K$ nearest neighbors of $\mathbf{q}$ from *dbVal*;
for *each concept model* $o = 1, ..., O^i$ **do**
    - Compute $\tau_o^i(\mathbf{q})$, the percentage of correctly classified samples from Ψ by classifier IC_o^i;
end

4.5.4 Specific DMCE Implementation Parameters

4.5.4.1 Feature Extraction and Selection

For each dataset, to generate the sets of reference ROI patterns $\mathbf{A}^i[t]$ from the reference trajectories $T[t]$ during training, as well as the input patterns $\mathbf{q}$ during operations, features are extracted using the Local Binary Pattern (LBP) (Ahonen *et al.*, 2006) algorithm, only considering the 59 *uniform patterns*. Each one of these $D = 59$ features is normalized between 0 and 1 using min-max scaling.

4.5.4.2 Classifier Training and Ensemble Prediction

In the same way than the system presented in (Pagano *et al.*, 2014), the proposed DMCE is implemented using Probabilistic Fuzzy-ARTMAP (PFAM) (Lim and Harrison, 1995) classifiers, trained and optimized with the Dynamic Particle Swarm Optimization training strategy presented in (Connolly *et al.*, 2012) and (Pagano *et al.*, 2014), using the Dynamic Niching PSO (DNPSO) variant (Nickabadi *et al.*, 2008a). The learning strategy is initialized with a swarm of 50 particles, 6 sub-swarms of maximum 5 particles, a maximum of 30 iterations and an early stopping criterion of no fitness improvement for 5 consecutive iterations. After convergence, the global best particle along with the 6 local bests associated to each sub-swarm are added to the final pool, each one associated with the corresponding concept. For each pool EoC^i and

each input pattern $\mathbf{q}$, the prediction $p^i(\mathbf{q})$ is produced using individual specific thresholds on the combined scores, following $p^i(\mathbf{q}) = S^i(\mathbf{q}) >= \Theta^i$, with Θ^i the individual-specific decision threshold selected during validation for a false alarm rate under 5%.

4.5.4.3 Concept Densities

Fuzzy C-Means clustering is implemented with a standard fuzziness parameter of $m = 2$. When a new set of reference ROI patterns $\mathbf{A}^i[t]$ is available, FCM clustering is performed to compute the concept model $\mathscr{A}^i[t] = \{\mu^i_{A,l'}[t], n^i_{A,l'}[t]\}$, using the internal Calinski-Harabasz validation measure to determine the optional number of cluster L'. Clustering is validated for $L' = 1$ to a maximum of $L' = 20$ clusters, with an early stopping criterion of no improvement for 5 consecutive iterations. The weights $n^i_{A,l'}[t]$ ($l' = 1,...,L'$) are determined by associating each sample to the closest center.

4.5.4.4 Tracking and Accumulation of Predictions

Fusion of predictions from the individual's ensemble of classifier is accomplished via evidence accumulation, emulating the brain process of working memory (Barry and Granger, 2007). For each initiated track $r = \{1,...,R\}$ and for each consecutive ROI $\mathbf{q}_j$ ($j = 1,...,J$) associated with this track, EoC^i generates a binary prediction $p^i(\mathbf{q}_j)$ (true, the individual is recognized, or false). The accumulated decision is computed with a moving overlapping window of size $W = 30$ ROIs (1 second in a 30 frames per second video), following $d^i_r = \sum^j_{j'=j-W} p^i(\mathbf{q}_{j'})$ The presence of the individual i in the track r can be confirmed if the accumulated response goes over a user-defined maximum number of consecutive activations.

4.5.5 Protocol for Validation

The same protocol is considered than in (Pagano *et al.*, 2014) and (Pagano *et al.*, 2015). For each time step t, and each individual $i = 1,...,10$, a temporary dataset $dbLearn^i$ is generated, and used to perform training and optimization of the classifiers. It is composed of ROI pat-

terns (after feature extraction and selection) from $T^i[t]$, as well as twice the same amount of non target patterns equally selected from the UM dataset and the Cohort Model (CM) of the individual (patterns from the other individuals of interest). Selection of non target pattern is performed using the *Condensed Nearest Neighbor* (CNN) algorithm (Hart, 1968). About the same amount of target and non-target patterns is generated using CNN, as well as the same amount of patterns not selected by the CNN algorithm, in order to have patterns close to the decision boundaries between target and non-target, as well as some patterns corresponding to the center of mass of the non target population.

The experimental protocol follows the (2x5 fold) cross-validation process to produce 10 independent replications, with pattern order randomization at the 5th replication. For each independent replication, $dbLearn^i$ is divided into the following subsets based on the 2x5 cross-validation methodology (with the same target and non-target proportions): (1) $dbTrain^i$ (2 folds): the training dataset used to design and update the parameters of the PFAM classifiers IC^i_o ($o = 1,...,L^i_o$), (2) ${dbVal_{ep}}^i$ (1 fold): the first validation dataset used to select the number of PFAM training epochs (the amount of presentations of patterns from $dbTrain^i$ to the networks) during the DNPSO optimization, and (3), STM^i (2 folds): the second validation dataset, used, to perform the DNPSO optimization. Using recommended parameters in (Connolly *et al.*, 2012), an incremental learning strategy based on DNPSO is then employed to conjointly optimize all parameters of these classifiers (weights, architecture and hyper-parameters) such that the area under the ROC curve is maximized.

When a gradual change is detected, and a previously-learned concept is updated, an existing swarm of classifiers is re-optimized using the DNPSO training strategy. The optimization resumes from the last state – the parameters of each classifier of the swarm. On the other hand, when an abrupt change is detected, a completely new swarm is generated and optimized for the new concept $\Omega^i_{O^i}$. The classifiers from each concept are then combined into $EoC^i = \{IC^i_1,...,IC^i_{O^i}\}$, and a validation ROC curve is produced from validation data from all concepts, from which the class specific threshold Θ^i is selected satisfying the constraint fpr $\leq 5\%$ for the highest tpr value.

For the DMCE system using DS-LA dynamic selection, data in STM^i are saved and used as a validation dataset for dynamic selection. With the reference VSkNN system, STM^i is used to validate the k parameter. In the same way, the k parameter of TCM-kNN is validated using STM^i, along with the system's internal thresholds. Finally, the ASR system is trained in batch mode, with $dbLearn^i$ accumulating at each time step. For every system, the class specific threshold Θ^i is validated in the same way than the proposed DMCE, with the exception of the ASR system that directly produces decisions.

4.5.6 Performance Evaluation

System performance is evaluated at two levels:

a. *Transaction* level, where the testing dataset is presented one ROI at a time, and individual predictions considered for performance evaluation.

b. *Trajectory* level, where the testing dataset is presented one sequence at a time, for each individual in the database. A perfect tracker is considered, where each session of each individual represents a different track on which predictions are accumulated for performance evaluation.

4.5.6.1 Transaction-level performance

To measure system performance, the classifiers are characterized by their true positive rate (tpr) and false positive rate (fpr), respectively the proportion of positives correctly classified over the total number of positive samples, and the proportion of negatives incorrectly classified (as positives) over the total number of negative samples. While these measurements are related to the operating point represented by the selected thresholds Θ^i, global performance is also represented with a ROC curve, a parametric plotting tpr against fpr for all possible threshold values. More precisely, the area under the ROC curve (AUC) or the partial AUC (for a range of fpr values) have been largely suggested as a robust scalar summary of 1- or 2-class classification

performance. To focus on a specific part of the ROC curve, the partial AUC $pAUC$ for fpr $\leq 5\%$ is considered for global performance estimation. In a video-surveillance application, non-target individuals are often much greater than the target ones. ROC measure may be inadequate as it becomes biased towards the negative class (Weiss, 2003). For this reason, the precision-recall space has been proposed to remain sensitive to this bias. Indeed, the precision is defined as the ratio $TP/(TP+FP)$ (with TP and FP the number of true and false positives), and the recall is an another denomination of the tpr. Precision allows to assess the accuracy for target patterns. The precision and recall measures can be summarized by the F_1 scalar measure, which can be interpreted as the harmonic mean of precision and recall.

4.5.6.2 Trajectory-level performance

For each individual, the predictions are accumulated with a moving window of $W = 30$ ROIs in a trajectory. The individual is detected when the accumulated activation go past a defined threshold. To assess the overall performance of the different systems for every individual i, an overall accumulation ROC curve is generated, with accumulation threshold going from 0 to 30 (the size of the moving window). For each target sequence, a true positive occurs when the maximum value of the accumulated predictions goes over the threshold In the same way, a false positive occurs when the maximum value of the accumulated predictions for non-target sequences goes over the threshold. To summarize the system performances, the $pAUC$ of the overall accumulated ROC curves is used as with the transaction-level measures. tpr and fpr measures are also presented, for an accumulation threshold of half the size of the accumulation window, 15 samples.

4.6 Results and Discussion

4.6.1 Transaction-level Performance

Average transaction-level performance for the the 10 individuals of interest are presented in Figure 4.4(a) and (b) for the FIA dataset scenario, and Figure 4.4(c) and (d) for the ChokePoint

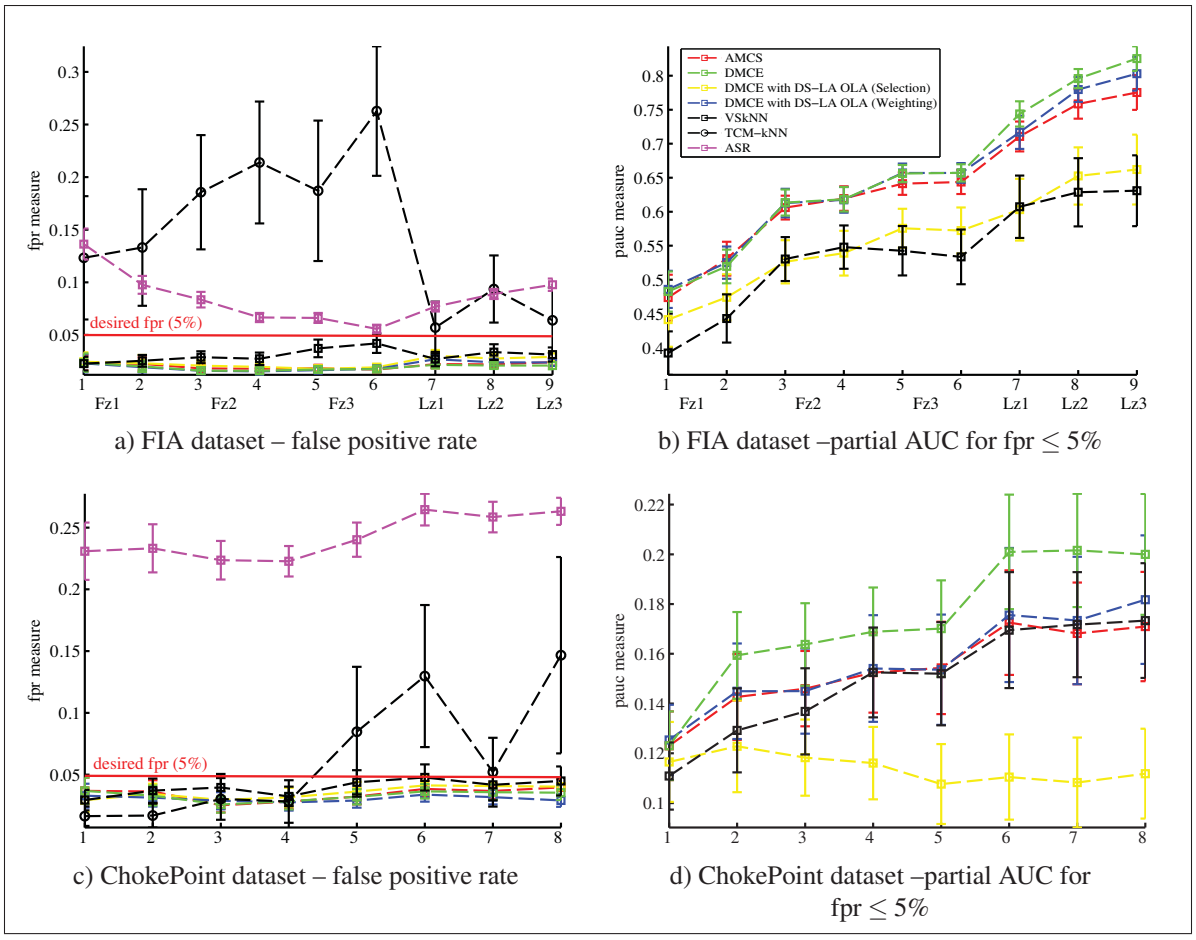


Figure 4.4 Average transaction-level classification performance for the 10 individuals of interest.

dataset scenario, with fpr and global *pAUC* (for fpr $\leq 5\%$) measures. First of all, it can be observed that TCM-kNN and ASR systems' fpr remains above the 5% validation constraint during the majority of the simulation for both datasets. For TCM-kNN with ChokePoint, the fpr increase starts at $t = 5$, which correspond to the introduction of reference samples from portal 2 that triggered an abrupt change detection for 8 individuals of interest out of 10. To simplify the figures, only the performance of the other 5 systems that respected the fpr $\leq 5\%$ constraint are presented for *pAUC* (Fig. 4.4(b)-(d)).

4.6.1.1 FIA Dataset

In Figure 4.4 (b), it can be observed that VSkNN's *pAUC* performance remains below AMCS, DMCE and DMCE with DS-LA OLA weighting for the entire simulation, at comparable level to DMCE with DS-LA OLA selection from $t = 3$ to $t = 9$. $t = 3$ corresponds to the introduction of reference captures from the second FIA enrollment session, which triggered an abrupt change detection for 8 out of 10 target individuals, and thus the addition of new classifiers to the ensembles. In this scenario, system performance is tested with images captured with a different focal length than reference captures, simulating a real-life VS scenario with variability in image resolution (e.g. due to subjects distance from the camera) and thus observed concepts. In this context, only selecting a single best classifier based on validation data representing different concepts provided a lower performance than relying on entire ensembles, as their internal diversity enables a higher flexibility w.r.t. changing concepts.

While other systems exhibit comparable performance from $t = 3$ to $t = 6$, a general increase can be observed at $t = 7$, which corresponds to the introduction of reference captures with different facial poses, and triggered an abrupt change detection for all target individuals. AMCS, DMCE with DS-LA OLA weighting and DMCE then show a performance increase from $t = 7$ to $t = 9$,

In this scenario, the last 3 sequences that introduced significantly different concepts for every individual put an emphasis on the benefits of the proposed dynamic score-level fusion. Weighting each classifier's contribution by its estimated competence using DMCE's framework provides a higher performance boost than other ensemble-based systems. In addition to exhibiting higher performance than the DS-LA OLA weighting, DMCE's fusion fusion only requires an average of 20 distance measures per prediction (average amount of cluster centers for each individual) at $t = 9$, as opposed to DS-LA OLA that requires an average of 292 (average amount of reference sample in validation datasets).

4.6.1.2 ChokePoint Dataset

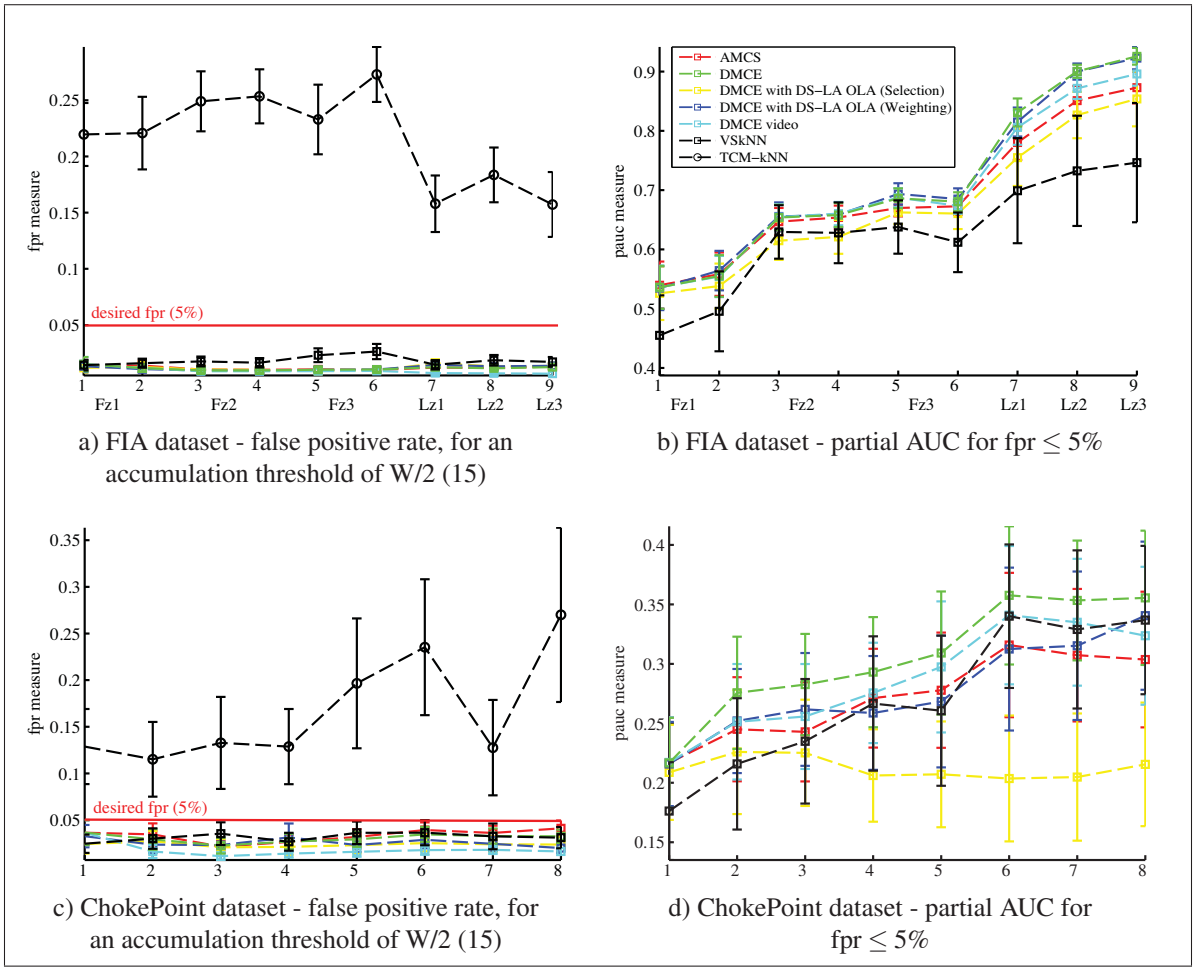
First of all, lower *pAUC* performance measures for every system (Fig. 4.4 (d)) indicate that the ChokePoint scenario represents a significantly harder FR problem than FIA, which can be explained by a higher variability in capture sessions (natural walking, different portals, entering/leaving sequences, etc.).

A similar behavior than with the FIA dataset can be observed with *pAUC* measures of DMCE with DS-LA OLA selection, that remain below every other system for the entire simulation. In addition, although VSkNN starts at a lower *pAUC*, DMCE with DS-LA OLA weighting, VSkNN and AMCS exhibit similar *pAUC* performance from $t = 3$ to $t = 8$. Finally, DMCE exhibits significantly higher *pAUC* performance than every other system from $t = 2$, where new classifiers are introduced due to abrupt change detection for 9 individuals out of 10, until $t = 8$, where it ends at 0.20 ± 0.02.

In the same way than the FIA scenario, the ChokePoint scenario puts an emphasis on the benefits of DMCE's dynamic score-level fusion. As soon as new classifiers are added to the system, it provides a higher performance boost that is maintained for the remaining of the simulation. In addition, it provides significantly higher performance than DS-LA OLA weighting, that remains at a similar level than AMCS relying on average score-level fusion.

As opposed to the FIA scenario, the systems are tested with captures from different enrollment sessions than reference captures, which favors the presence of unknown concepts during testing, not represented in the validation set of the DS-LA OLA methods nor in classifier parameters. In such cases, classifiers are likely to exhibit similar behavior, which generates equivalent fusion weights, and thus mimics the behavior of a standard average fusion. On the other hand, cluster-based concept representations allow DMCE to rely on an additional source of information to fuse classifiers' outputs, without the bias introduced by their training. This higher level of specialization enables to produce higher performance as soon as different classifiers are added to ensembles.

4.6.2 Trajectory-level Performance



a) FIA dataset - false positive rate, for an accumulation threshold of W/2 (15)

b) FIA dataset - partial AUC for fpr $\leq 5\%$

c) ChokePoint dataset - false positive rate, for an accumulation threshold of W/2 (15)

d) ChokePoint dataset - partial AUC for fpr $\leq 5\%$

Figure 4.5 Average trajectory-level classification performance for the 10 individuals of interest.

4.6.2.1 FIA Dataset

Average trajectory-level performance for the the 10 individuals of the FIA dataset scenario are presented in Figure 4.5(a) and (b). While a similar trend than with the transaction-level can be observed, with DMCE and DMCE with DS-LA OLA weighting providing a higher performance level than every other system from session 7 to 9, the temporal accumulation mechanism enables DMCE with DS-LA OLA selection to exhibit similar classification performance than

AMCS for every performance measure. It can also be noted that accumulation increases the performance of every system, with DMCE ending at a *pAUC* of 0.93 ± 0.01.

An additional DMCE variant is also presented in Figure 4.5, called DMCE video. A similar dynamic score-level fusion is considered, but instead of computing fusion weights for each capture, a unique set of weights is computed for each sequence, using the sequence-to-model distance presented in Equation 4.1 of Section 4.4.3. It can be observed that DMCE video produces a lower performance than DMCE, ending at a *pAUC* of 0.90 ± 0.02, closer to AMCS' performances. This behavior confirms that, in VS, concept variations can be observed within facial trajectories as observation conditions evolve. For this reason, a dynamic decision fusion mechanism that adapts to each capture provides a higher precision for competence evaluation, and thus overall system performance.

4.6.2.2 ChokePoint Dataset

Average trajectory-level performance for the the 10 individuals of the ChokePoint dataset scenario are presented in Figure 4.5(c) and (d). In the same way than with the FIA dataset, a similar trend than with the transaction-level performance can be observed. DMCE shows higher *pAUC* performance for the entire simulation, ending at 0.36 ± 0.06 (v.s 0.20 ± 0.02 without accumulation).

In addition, DMCE video also produces a lower *pAUC* performance than DMCE.

4.7 Conclusion

This paper presents a new framework for video-to-video face recognition using adaptive ensembles of classifiers. Called Dynamic Multi-Concept Ensemble (DMCE), it is comprised of a pool of incremental learning classifiers for each individual registered to the system, where each classifier is specialized in different capture conditions detected in reference trajectories. During enrollment and update phases, these capture conditions are gradually modeled as multi-modal concept densities, through on-line clustering of reference trajectories. During operations, these

densities are used to evaluate the competence of each classifier for any given facial capture. This allows for a dynamic adaptation of the ensembles' fusion functions, weighting each classifier according to its relevance for the observed capture condition.

A particular implementation of DMCE is proposed for proof-of-concept experiments. For each enrolled individual, it is comprised of a pool of 2-class Probabilistic Fuzzy-ARTMAP classifiers, generated and updated using an incremental training strategy based on a dynamic Particle Swarm Optimization. Concept densities are represented as Fuzzy C-means centers, and classifier competence is estimated from Fuzzy C-means degrees of belonging. These measures are then used for weighted average score-level fusion. Simulation results indicate that the DMCE framework allows to maintain a high level of performance when facial models are gradually generated using significantly different reference trajectories. DMCE's dynamic decision fusion produces a higher classification performance than reference dynamic selection methods, for a significantly lower computational complexity. In addition, the proposed implementation provides a higher performance than a probabilistic kNN based system adapted to video-to-video FR, a reference open-set TCM-kNN system as well as an Adaptive Sparse Representation face recognition system.

In this paper, a key assumption for DMCE is that facial models of individuals are comprised of multiple concepts that can be learned from different reference trajectories. However, in real-life surveillance applications, facial models are likely to evolve and change over time, with concepts becoming irrelevant as the individuals age or change their appearance. For future work, a practical implementation of DMCE would require a bounding of its computational complexity, for example through the development of purging strategies to remove concepts becoming irrelevant over time.

CONCLUSION AND RECOMMENDATIONS

Face recognition is becoming increasingly popular in security applications such as in intelligent video surveillance, as it provides significant advantages over other biometric modalities (e.g. iris or fingerprint). Specifically, the ability to capture faces in a dense crowd without requiring any cooperation from observed individuals may be critical in many scenarios, such as *watch list screening*, *person re-identification* and *search and retrieval* in archived videos. However, face recognition in video surveillance still faces many challenges, as the faces captured during operations exhibit significant variations due to the lack of control over observation conditions. As face recognition system are typically designed a priori using a limited amount of reference captures, the facial models used for detection and often poor representatives of the complexity of the recognition problem. Furthermore, these systems are operated in changing environments, in which they need to quickly adapt to these changes to remain accurate.

In this thesis, a new framework for a dynamic face recognition system for video surveillance is proposed. It allows to continuously update facial models of individuals of interest as new reference videos become available, as well as dynamically adapt its behaviour to changing observation conditions in operations. The key component is the representation of different observation conditions encountered in reference videos, which allows to improve system performance through the adaptation of both learning and operational strategies to the nature of the observed environment. Concept models are used during training and update of facial models, to ensure that new concepts are assimilated without corrupting previously-acquired knowledge by guiding an hybrid ensemble updating strategy. In addition, these models are used during operations, to estimate each classifier's relevant w.r.t. each operational capture, dynamically adapting ensembles' fusion rule to changing observation conditions.

In Chapter 2, *concept change* detection was first considered to mitigate the growth in complexity of a *self-updating* face recognition system over time. A *context-sensitive* self-updating

technique has been proposed for *template matching* systems, in which galleries or reference *ROI patterns* are only updated with highly-confident captures exhibiting significant changes in capture conditions. Proof of concept experiments have been conducted with a standard *template matching* system detecting changes in *illumination conditions*, using thee publicly-available face databases. This technique enabled to maintain the same level of performance than a regular *self-updating template matching* system, while reducing the size of template galleries by half.

In Chapter 3, an adaptive *multi-classifier system* was proposed for face recognition in video surveillance. It is comprised of an ensemble of incremental classifiers per enrolled individuals, and allows to refine facial models with new reference data available over time. To assimilate new concepts while preserving previously-acquired knowledge, this system relies on *concept change* detection to guide an hybrid learning strategy, mixing incremental learning with ensemble generation. For each individual, new classifiers are only added to their specific ensembles when an abrupt change is detected in reference data. On the other hand, when a gradual change is detected, knowledge about corresponding concepts is refined through incremental learning of classifiers. A particular implementation has been proposed, using ensembles of probabilistic Fuzzy-ARTMAP classifiers generated and updated with dynamic Particle Swarm Optimization, and the Hellinger Drift Detection Method for change detection. Experimental simulations with the FIA video surveillance database indicated that the proposed active methodology allowed to mitigate the effects of knowledge corruption, exhibiting higher classification performance than a similar passive system, and reference probabilistic kNN and TCM-kNN systems.

Finally, in Chapter 4, an evolution of the framework presented in Chapter 3 was proposed, adding the ability to adapt system behaviour to changing operating conditions. A new dynamic weighting fusing rule was proposed for ensembles of classifiers, where classifier competences are estimated from multi-modal concept densities used for *concept change* detection during

training. An evolution of the particular implementation presented in Chapter 3 has been presented, where concept densities are estimated with the Fuzzy C-Means clustering algorithm, and ensemble fusion is performed through dynamic weighted score-average. Experimental simulations with the FIA and ChokePoint video-surveillance datasets showed that the proposed dynamic fusion method enabled to provide a higher classification performance than the DS-OLA dynamic selection method, for a significantly lower computational complexity. In addition, the proposed system exhibited higher performance than reference probabilistic kNN, TCM-kNN and Adaptive Sparse Coding systems.

Future Work

The final iteration of the proposed framework allows to increase face recognition performance in video surveillance by refining facial models over time and dynamically adapt its behaviour during operations, but its mechanisms involve an increase of computational and memory complexity over time. Although the addition of new classifiers is controlled by concept change detection, a real-life implementation would require to develop additional strategies to maintain system complexity at acceptable levels (e.g. for live detection to remain possible). For example, pruning strategies may be investigated to regularly remove concepts that become obsolete over time due to the aging of individuals, or other permanent changes in facial appearance.

In addition, this thesis only considered the use of faces to perform recognition, but other biometric modalities such as gait can be analyzed from video sequences. Following the same principles than *co-updating* techniques for *semi-supervised* learning, system performance could be increased by combining multiple modalities in ensembles of classifiers, and sharing information about changes in the environment.

Finally, the lack of representative reference data may be addressed through sample generation and information sharing among individuals. When specific concepts are lacking in certain indi-

viduals' reference data (e.g. a specific facial pose), synthetic *ROI patterns* could be generated from concept models of other individuals, for example with virtual pose generation techniques.

APPENDIX I

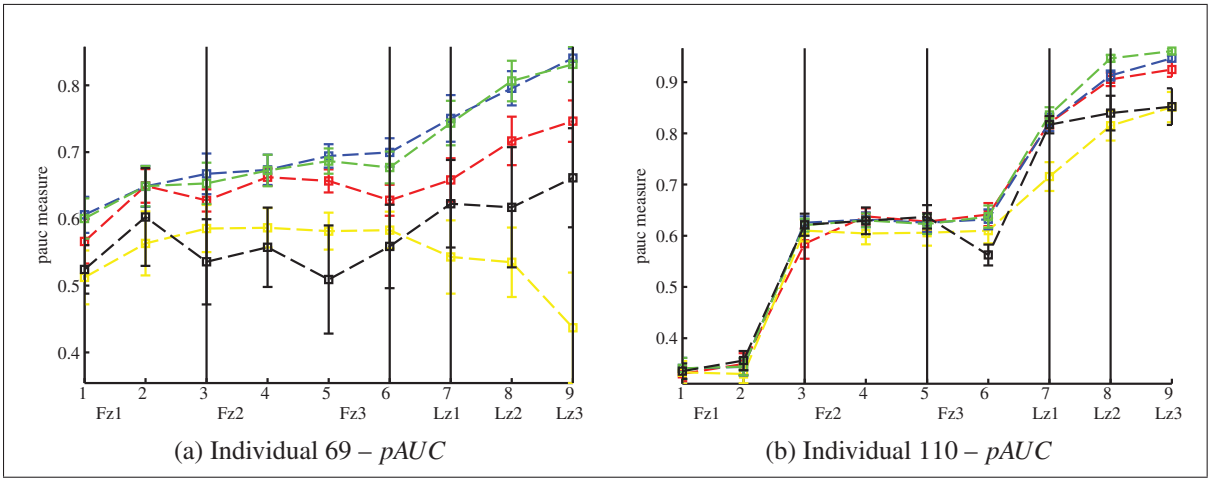
SUPPLEMENTARY RESULTS FOR DMCE IMPLEMENTATION (CHAPTER 4)

1. Detailed Performance Analysis for Two Specific Individuals

The cases of individuals 69 and 110 of the FIA dataset is of a particular interest for a deeper analysis of system performance, as they respectively represent *good* and *bad* cases for DMCE.

1.1 Transaction-level Performance

pAUC performance measures for these two individuals are presented in Figure I-1, with vertical black bars indicating changes detected in reference data.



(a) Individual 69 – *pAUC*

(b) Individual 110 – *pAUC*

Figure-A I-1 Transaction-level *pAUC* performance for individuals 69 and 110 of the FIA dataset, respectively representing a *good* and a *bad* case.

As a matter of fact, individual 69 can be considered as a *good* case, where dynamic weighting (through either DMCE's dynamic weighting or DS-LA OLA) produces a significant performance boost from $t = 6$, where change is detected, and new classifiers added to the ensemble to integrate knowledge about additional frontal and profile captures. A deeper understanding

of these different behaviors can be provided by the analysis of the concept models detected for these two individuals, represented in 2D in Figure I-2.

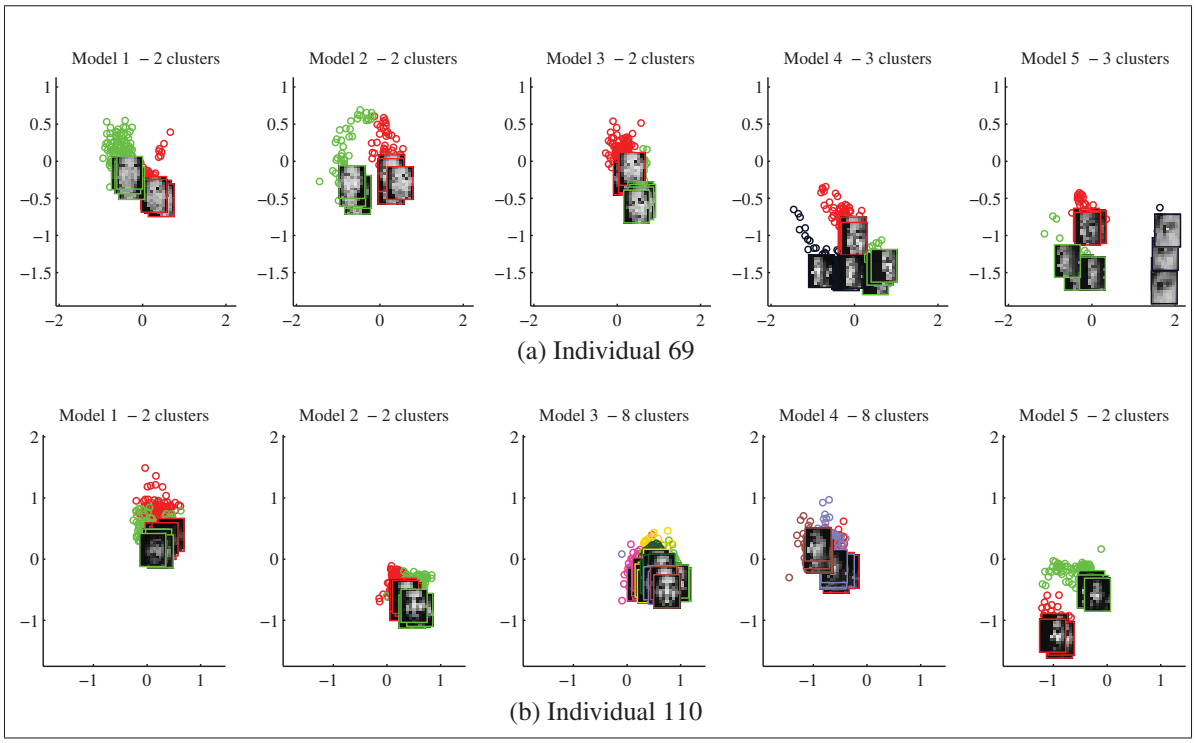


Figure-A I-2 2D projections of the 5 concept densities generated by the change detection module of DMCE for individual 69 and 110 of the FIA dataset Goh *et al.* (2005), using Sammon mapping.

As a matter of fact, with individual 69, clusters generated for concept models corresponding to frontal (models 1 to 3) and profile (models 4 and 5) remain close in the Sammon space, as opposed to models of individual 110 that are visibly distinct. The latter case is significantly easier, as shown by the performance mesures in Figure I-1, as the visible distance between each concept is more likely to generate classifiers with distinct decision boundaries, thus reacting differently to each concept. The competence measure would then only echo this behavior, assigning lower weights to classifiers that produce lower scores, and vice-versa. In such case, DMCE's fusion behavior becomes similar to a standard average, as can be observed in Figure I-1. On the other hand, with individual 69, concept models provide a more precise represen-

tation of the intra-class variability of the underlying data distribution than classifiers decision boundaries, and this additional source of information allows the refine the fusion process.

1.2 Trajectory-level Performance

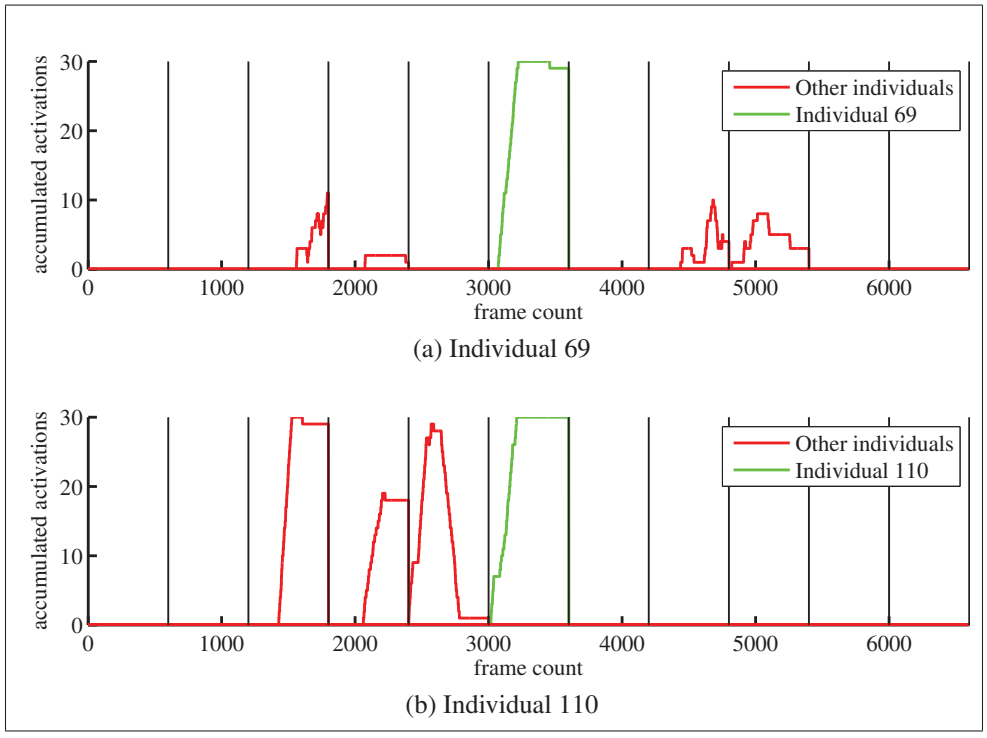


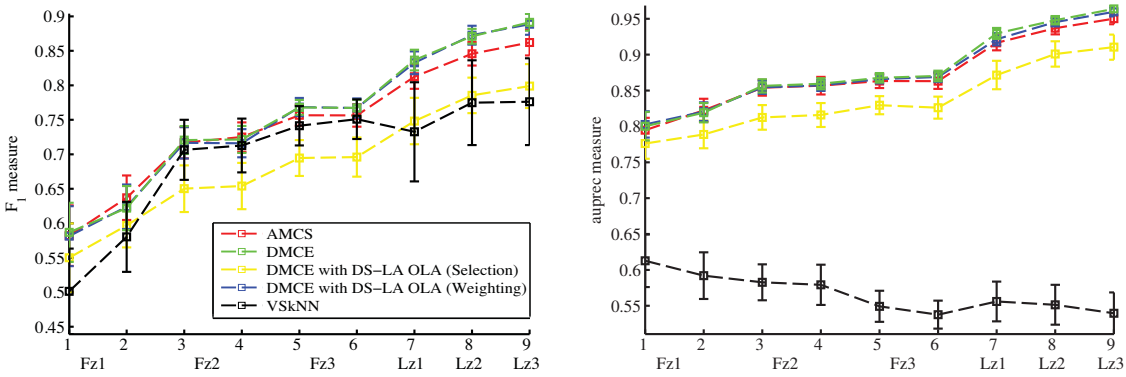
Figure-A I-3 Accumulated activations of classification modules for individual 69 ((a)) and 110 ((b)) of the FIA dataset. A custom scenario is considered, where ten sequences of non-target individuals (in red) are concatenated with one sequence of the target individual (in green), extracted from the first FIA enrollment session. Each sequence is 600 frames long, and is delimited by vertical black bars.

Figure I-3 presents accumulated activations for the trajectory-level analysis of the performance for both individuals. In each cases, a custom scenario is designed, concatenating ten sequences of non target individuals (picked at random) with one sequence of the target individual from the first FIA session (600 frames each). As the sequences usually end abruptly with the individual still present in the scene, abrupt drops in accumulation activation can be observed when a a new accumulation is initiated for the following trajectory. Figure I-3 (a) illustrate a *good* case

with individual 69, where non-target accumulations do not go over 12 activations, and target accumulation quickly reach the maximum of 30 (size of the window). On the other hand, I-3 (b) illustrate false positives, where two sequences of a non-target individual managed to reach 30 accumulated activations.

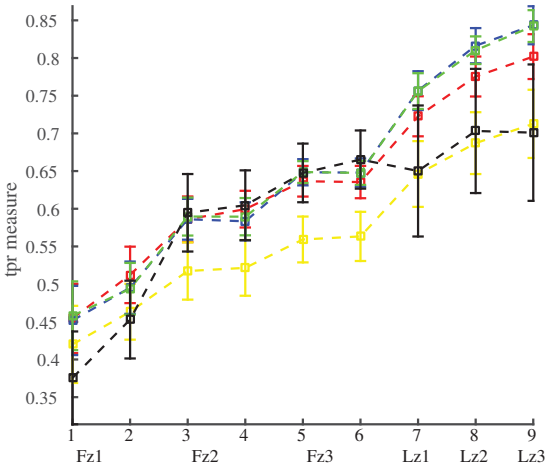
2. Additional Performance Measures

Figure I-4 presents additional transaction-level performance measures for both datasets (F_1, area under the p-roc curve and tpr), and Figure I-5 additional trajectory-level performances measures (F_1, area under the p-roc curve and tpr).

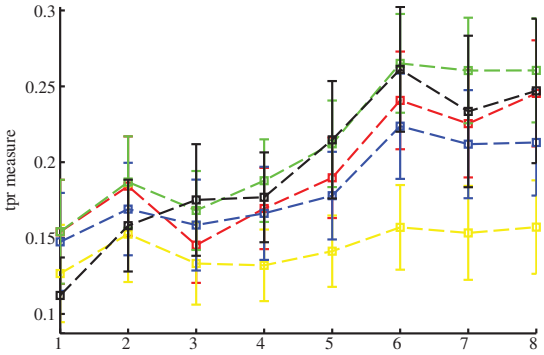


(a) FIA dataset – F_1 measure for the selected operating point

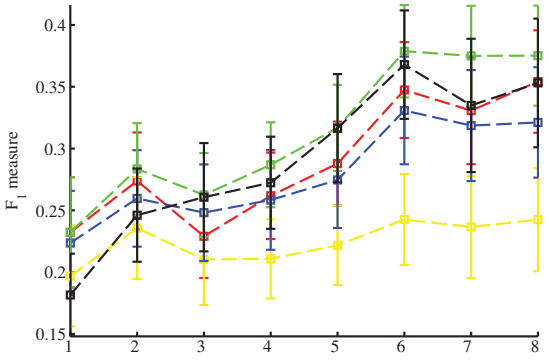
(b) FIA dataset – area under the p-roc curve



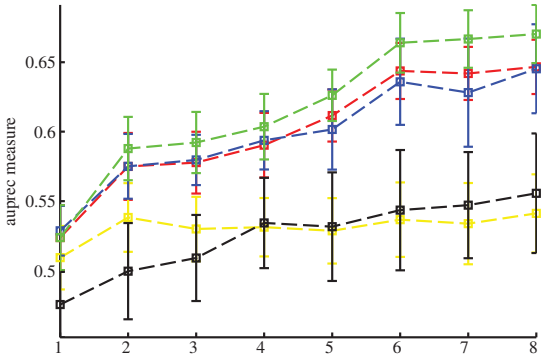
(c) FIA dataset – true positive rate for the selected operating point



(d) ChokePoint dataset – true positive rate for the selected operating point

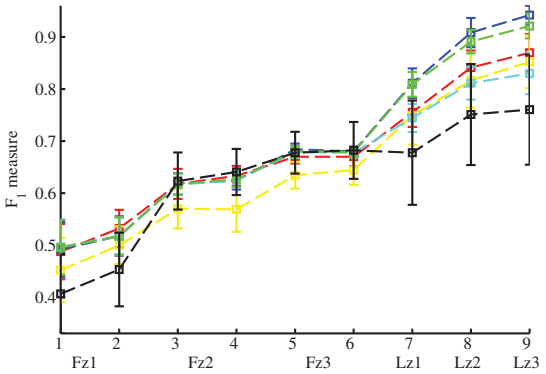


(e) ChokePoint dataset – F_1 measure for the selected operating point

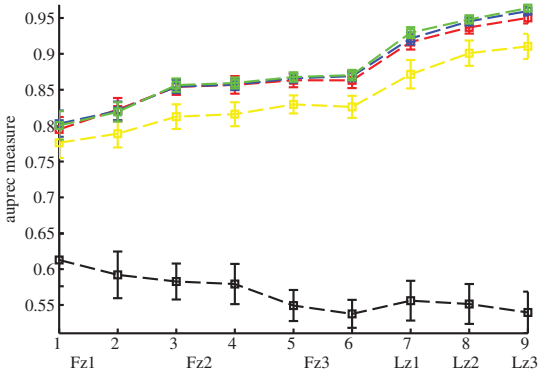


(f) ChokePoint dataset – area under the p-roc curve

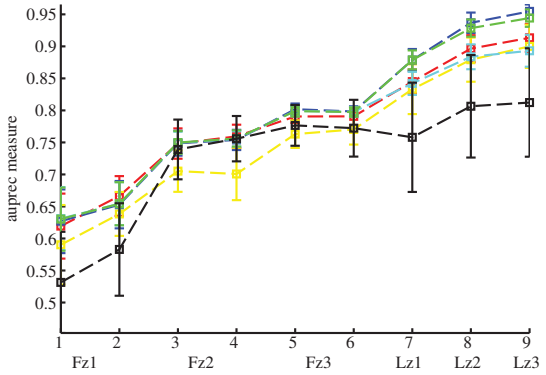
Figure-A I-4 Additional transaction-level classification performance measures for the 10 individuals of interest.



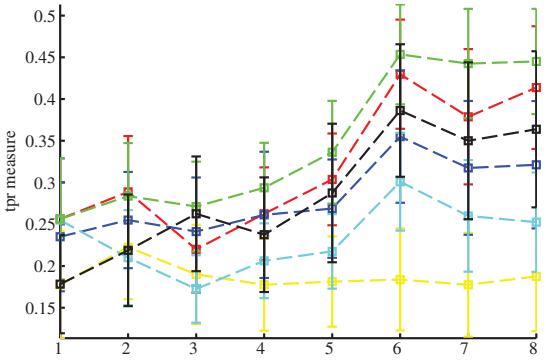
(a) FIA Dataset – F_1 measure, for an accumulation threshold of W/2 (15)



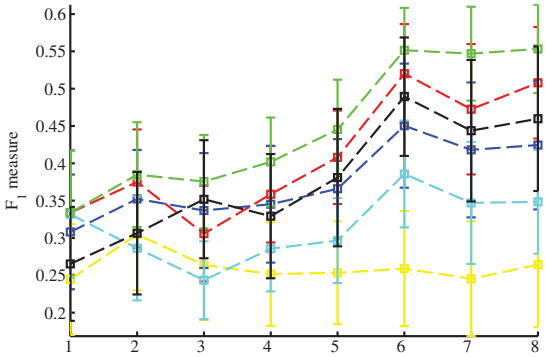
(b) FIA dataset – area under the p-roc curve



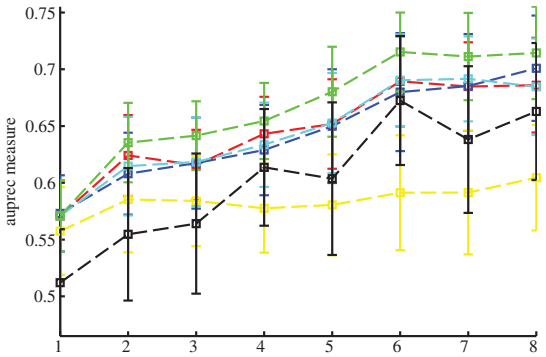
(c) FIA Dataset – true positive rate for an accumulation threshold of W/2 (15)



(d) ChokePoint dataset – true positive rate for an accumulation threshold of W/2 (15)



(e) ChokePoint Dataset – F_1 measure, for an accumulation threshold of W/2 (15)



(f) ChokePoint dataset – area under the p-roc curve

Figure-A I-5 Additional trajectory-level performance classification performance measures for the 10 individuals of interest.

BIBLIOGRAPHY

Ahonen, T., A. Hadid, and M. Pietikainen. 12 2006. "Face description with local binary patterns: application to face recognition". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, n° 12, p. 2037–41.

Alippi, C., G. Boracchi, and M. Roveri. 2013. "Just-In-Time Classifiers for Recurrent Concepts". *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, n° 4, p. 620 - 34.

Alippi, C., G. Boracchi, and M. Roveri. 2011. "A just-in-time adaptive classification system based on the intersection of confidence intervals rule". *Neural Networks*, vol. 24, n° 8, p. 791–800.

Ansari, A., M. Abdel-Mottaleb, et al. 2003. "3D face modeling using two views and a generic face model with application to 3D face recognition". In *Advanced Video and Signal Based Surveillance, 2003. Proceedings. IEEE Conference on*. p. 37–44. IEEE.

Barr, J. R., K. W. Bowyer, P. J. Flynn, and S. Biswas. 2012. "Face recognition from video: A review". *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 26, n° 05, p. 1266002.

Barry, M. and E. Granger. 2007. "Face recognition in video using a what-and-where fusion neural network". In *International Joint Conference on Neural Networks, 12-17 Aug. 2007*. (Ecole de Technol. Super., Montreal, Canada 2007), p. 2255–60. IEEE.

Bashbaghi, S., E. Granger, R. Sabourin, and G.-A. Bilodeau. 2014. "Watch-List Screening Using Ensembles Based on Multiple Face Representations". In *Pattern Recognition (ICPR), 2014 22nd International Conference on*. p. 4489–4494. IEEE.

Bengio, S. and J. Mariethoz. 2007. "Biometric person authentication is a multiple classifier problem". In *Multiple Classifier Systems. 7th International Workshop, MCS 2007. Proceedings, 23-25 May 2007*. p. 513-22. Springer.

Bezdek, J. C., R. Ehrlich, and W. Full. 1984. "FCM: The fuzzy c-means clustering algorithm". *Computers & Geosciences*, vol. 10, n° 2, p. 191–203.

Blackwell, T., J. Branke, et al. 2004. "Multi-swarm optimization in dynamic environments". In *EvoWorkshops*. p. 489–500. Springer.

Blanz, V. and T. Vetter. 2003. "Face recognition based on fitting a 3D morphable model". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, n° 9, p. 1063–1074.

Blum, A. 1997. "Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain". *Machine Learning*, vol. 26, n° 1, p. 5–23.

Brew, A. and P. Cunningham. 2009. "Combining cohort and UBM models in open set speaker identification". In *2009 Seventh International Workshop on Content-Based Multimedia Indexing (CBMI), 3-5 June 2009*. (Machine Learning Group, Univ. Coll. Dublin, Dublin, Ireland 2009), p. 62–7. IEEE.

Britto, A. S., R. Sabourin, and L. E. Oliveira. 2014. "Dynamic selection of classifiers—a comprehensive review". *Pattern Recognition*, vol. 47, n° 11, p. 3665–3680.

Brown, G., J. Wyatt, R. Harris, and Y. Xin. 2005. "Diversity creation methods: a survey and categorisation". *Information Fusion*, vol. 6, n° Copyright 2005, IEE, p. 5-20.

Carpenter, G. A., S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen. 1992. "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps". *IEEE Transactions on Neural Networks*, vol. 3, n° Copyright 1992, IEE, p. 698-713.

Carpenter, G. A. and S. Grossberg. 1987. "A massively parallel architecture for a self-organizing neural pattern recognition machine". *Comput. Vision Graph. Image Process.*, vol. 37, n° 1, p. 54–115.

Carpenter, G. A., S. Grossberg, and J. H. Reynolds. 1991. "ARTMAP. Supervised real-time learning and classification of nonstationary data by a self-organizing neural network". *Neural Networks*, vol. 4, n° 5, p. 565–588.

Chakraborty, D. and N. R. Pal. 2003. "A novel training scheme for multilayered perceptrons to realize proper generalization and incremental learning". *Neural Networks, IEEE Transactions on*, vol. 14, n° 1, p. 1–14.

Chandola, V., A. Banerjee, and V. Kumar. 2009. "Anomaly detection: A survey". *ACM computing surveys (CSUR)*, vol. 41, n° 3, p. 15.

Committee, W. B. et al., 2010. *Biometric Recognition:: Challenges and Opportunities*.

Connolly, J., E. Granger, and R. Sabourin. 2010a. "An Adaptive Ensemble of Fuzzy ARTMAP Neural Networks for Video-Based Face Classification". In *2010 IEEE Congress on Evolutionary Computation*. (Piscataway, NJ, USA 2010), p. 8 pp.–. IEEE.

Connolly, J.-F., E. Granger, and R. Sabourin. 2012. "An adaptive classification system for video-based face recognition". *Information Sciences*, vol. 192, p. 50 - 70.

Connolly, J.-F., E. Granger, and R. Sabourin. 2008. Supervised incremental learning with the fuzzy artmap neural network. *Artificial Neural Networks in Pattern Recognition*, p. 66–77. Springer.

Connolly, J.-F., E. Granger, and R. Sabourin. 2010b. "An adaptive classification system for video-based face recognition". p. –.

Connolly, J.-F., E. Granger, and R. Sabourin. 2013. "Dynamic multi-objective evolution of classifier ensembles for video face recognition". *Applied Soft Computing*, vol. 13, n° 6, p. 3149–3166.

Cootes, T. F., G. J. Edwards, and C. J. Taylor. 2001. "Active appearance models". *IEEE Transactions on Pattern Analysis & Machine Intelligence*, , p. 681–685.

De-la Torre, M., E. Granger, P. V. Radtke, R. Sabourin, and D. O. Gorodnichy. 2014. "Partially-supervised learning from facial trajectories for face recognition in video surveillance". *Information Fusion*.

De-la Torre, M., E. Granger, P. V. Radtke, R. Sabourin, and D. O. Gorodnichy. 2015. "Partially-supervised learning from facial trajectories for face recognition in video surveillance". *Information Fusion*, vol. 24, p. 31–53.

De Marsico, M., M. Nappi, D. Riccio, and H. Wechsler. 2012. "Robust Face Recognition for Uncontrolled Pose and Illumination Changes". *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 43, n° 1, p. 149–163.

Didaci, L., G. L. Marcialis, and F. Roli. 2014. "Analysis of unsupervised template update in biometric recognition systems". *Pattern Recognition Letters*, vol. 37, p. 151–160.

Ditzler, G. and R. Polikar. 2011. "Hellinger distance based drift detection for nonstationary environments". In *Symposium Series on Computational Intelligence, IEEE SSCI 2011 - 2011 IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments, CIDUE 2011, April 11, 2011 - April 15, 2011*. (Dept. of Electrical and Computer Engineering, Rowan University, Glassboro, NJ, United States 2011), p. 41–48. IEEE Computer Society.

Doc, I. 2005. "9303". *Machine Readable Travel Documents, Part*, vol. 2.

Doddington, G., W. Liggett, A. Martin, M. Przybocki, and D. Reynolds. 1998. *Sheep, goats, lambs and wolves: A statistical analysis of speaker performance in the nist 1998 speaker recognition evaluation*. Technical report.

Dos Santos, E. M., R. Sabourin, and P. Maupin. 2009. "Overfitting cautious selection of classifier ensembles with genetic algorithms". *Information Fusion*, vol. 10, n° 2, p. 150–162.

Dries, A. and U. Rückert. 2009. "Adaptive concept drift detection". *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 2, n° 5-6, p. 311–327.

Drummond, C. 2006. "Discriminative vs. generative classifiers for cost sensitive learning". In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. (Quebec City, Que., Canada 2006), p. 479–490. Springer Verlag.

Duda, R. O. and P. E. Hart. 1973. "Pattern recognition and scene analysis".

Eberhart, R. and J. Kennedy. 1995. "A new optimizer using particle swarm theory". In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 4-6 Oct. 1995*. (Purdue Sch. of Eng. Technol., Indianapolis, IN, USA 1995), p. 39–43. IEEE.

Ekenel, H., L. Szasz-Toth, and R. Stiefelhagen. 2009. "Open-Set Face Recognition-Based Visitor Interface System". In *Computer Vision Systems. Proceedings 7th International Conference, ICVS 2009*. (Berlin, Germany 2009), p. 43–52. Springer Verlag.

Freni, B., G. L. Marcialis, and F. Roli. 2008. Replacement algorithms for fingerprint template update. *Image Analysis and Recognition*, p. 884–893. Springer.

Freund, Y. and R. E. Schapire. 1996. "Experiments with a new boosting algorithm". In *Proceedings of Thirteenth International Conference on Machine Learning, 3-6 July 1996*. (ATT Bell Labs., Murray Hill, NJ, USA 1996), p. 148–56. Morgan Kaufmann Publishers.

Fritzke, B. 1996. "Growing self-organizing networks-why?". In *Proceedings of European Symposium on Artificial Neural Networks, 24-26 April 1996*. (Systembiophys., Ruhr-Univ., Bochum, Germany 1996), p. 61–72. D Facto.

Gama, J., P. Medas, G. Castillo, and P. Rodrigues. 2004. "Learning with drift detection". In *Advances in Artificial Intelligence - SBIA 2004. 17th Brazilian Symposium on Artificial Intelligence. Proceedings, 29 Sept.-1 Oct. 2004*. (LIACC, Porto Univ., Portugal 2004), p. 286–95. Springer-Verlag.

Goh, R., L. Liu, X. Liu, and T. Chen. 2005. "The CMU face in action (FIA) database". In *Analysis and Modelling of Faces and Gestures. Second International Workshop. AMFG 2005. Proceedings*. (Berlin, Germany 2005), p. 255–63. Springer-Verlag.

Gorodnichy, D. O. 2005a. "Video-based framework for face recognition in video". In *Proc. of 2nd Canadian Conference on Computer and Robot Vision*. p. 330–338.

Gorodnichy, D. 2005b. "Video-based framework for face recognition in video". In *Proceedings. The 2nd Canadian Conference on Computer and Robot Vision*. (Piscataway, NJ, USA 2005), p. 330–8. IEEE Comput. Soc.

Granger, E., J.-F. Connolly, and R. Sabourin. 2008. "A comparison of fuzzy ARTMAP and Gaussian ARTMAP neural networks for incremental learning". (Piscataway, NJ, USA 2008), p. 3305 - 12.

Grossberg, S. 1988. "Nonlinear neural networks: principles, mechanisms, and architectures". *Neural Networks*, vol. 1, n° 1, p. 17 - 61.

Hart, P. E. 1968. "The condensed nearest neighbor rule". *IEEE Transactions on Information Theory*, vol. 14, p. 515–516.

Ho, T. K. 08 1998. "The random subspace method for constructing decision forests". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, n° 8, p. 832–44.

Holmes, C. and N. Adams. 2002. "A probabilistic nearest neighbour method for statistical pattern recognition". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 64, n° 2, p. 295–306.

Jacobs, R. A., M. I. Jordan, S. J. Nowlan, and G. E. Hinton. 1991. "Adaptive mixtures of local experts". *Neural computation*, vol. 3, n° 1, p. 79–87.

Jafri, R. and H. Arabnia. 2009. "A Survey of Face Recognition Techniques". *Journal of Information Processing Systems*, vol. 5, n° 2, p. 41–68.

Jiang, X. and W. Ser. 2002. "Online fingerprint template improvement". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, n° 8, p. 1121–1126.

Kamgar-Parsi, B., W. Lawson, and B. Kamgar-Parsi. 2011. "Toward Development of a Face Recognition System for Watchlist Surveillance". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, n° 10, p. 1925 - 37.

Kapp, M. N., C. O. d. A. Freitas, and R. Sabourin. 2007. "Methodology for the design of NN-based month-word recognizers written on Brazilian bank checks". *Image and Vision Computing*, vol. 25, n° 1, p. 40–49.

Kittler, J. and F. M. Alkoot. 2003. "Sum versus vote fusion in multiple classifier systems". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, n° 1, p. 110–115.

Klingenberg, R. and I. Renz. 1998. "Adaptive information filtering: learning in the presence of concept drift". In *Proceedings of AAAI/ICML-98 workshop on learning for text categorization, Madison, WI*. p. 33–40.

Kuncheva, L. I. 2004a. "Classifier ensembles for changing environments". In *Multiple Classifier Systems. 5th International Workshop, MCS 2004. Proceedings, 9-11 June 2004*. (Sch. of Informatics, Wales Univ., UK 2004), p. 1–15. Springer-Verlag.

Kuncheva, L., 07 2004b. *Combining Pattern Classifiers: Methods and Algorithms*.

Kuncheva, L. I. 2008. "Classifier ensembles for detecting concept change in streaming data: Overview and perspectives". In *2nd Workshop SUEMA*. p. 5–10.

Kuncheva, L. I. 2009. "Using control charts for detecting concept change in streaming data". *Bangor University*.

Li, F. and H. Wechsler. 11 2005. "Open set face recognition using transduction". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, n° 11, p. 1686–97.

Li, S. Z. and A. K. Jain, 2011. *Handbook of Face Recognition*. ed. 2nd.

Lim, C. P. and R. F. Harrison. 1995. "Probabilistic Fuzzy ARTMAP: an autonomous neural network architecture for Bayesian probability estimation". In *Proceedings of 4th International Conference on Artificial Neural Networks (Conf. Publ. No.409), 26-28 June 1995*. (Sheffield Univ., UK 1995), p. 148–53. IEE.

Littlestone, N. 1988. "Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm". *Machine learning*, vol. 2, n° 4, p. 285–318.

Liu, X. and T. Cheng. 2003. "Video-based face recognition using adaptive hidden Markov models". In *CVPR 2003: Computer Vision and Pattern Recognition Conference, 18-20 June 2003*. (Electr. Comput. Eng., Carnegie Mellon Univ., Pittsburgh, PA, USA 2003), p. 340–5. IEEE Comput. Soc.

Liu, X., T. Chen, and S. M. Thornton. 2003. "Eigenspace updating for non-stationary process and its application to face recognition". *Pattern Recognition*, vol. 36, n° 9, p. 1945–1959.

Marcialis, G. L., F. Roli, and G. Fadda. 2014. "A novel method for head pose estimation based on the "Virtuvian Man"". *International Journal of Machine Learning and Cybernetics*, vol. 5, n° 11, p. 111-124.

Marcialis, G. L., A. Rattani, and F. Roli. 2008. Biometric template update: an experimental investigation on the relationship between update errors and performance degradation in face verification. *Structural, Syntactic, and Statistical Pattern Recognition*, p. 684–693. Springer.

Matta, F. and J. L. Dugelay. 2007. "Video face recognition. a physiological and behavioural multimodal approach". In *2007 IEEE International Conference on Image Processing, ICIP 2007, 16-19 Sept. 2007*. (Eurecom Inst., Sophia Antipolis, France 2007), p. 497–500. IEEE.

Matta, F. and J.-L. Dugelay. 2009. "Person recognition using facial video information: a state of the art". *Journal of Visual Languages and Computing*, vol. 20, n° 3, p. 180 - 7.

Mery, D. and K. Bowyer. 2014. "Recognition of Facial Attributes Using Adaptive Sparse Representations of Random Patches". In *Computer Vision-ECCV 2014 Workshops*. p. 778–792. Springer.

Minku, L. L., A. P. White, and X. Yao. 2010. "The impact of diversity on online ensemble learning in the presence of concept drift". *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, n° 5, p. 730–742.

Minku, L. and X. Yao. 2012. "DDD: A New Ensemble Approach for Dealing with Concept Drift". *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, n° 4, p. 619 - 33.

Muhlbaier, M. D., A. Topalis, and R. Polikar. 01 2009. "Learn++ .NC: combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes". *IEEE Transactions on Neural Networks*, vol. 20, n° 1, p. 152–68.

Muhlbaier, M. D. and R. Polikar. 2007. An ensemble approach for incremental learning in nonstationary environments. *Multiple classifier systems*, p. 490–500. Springer.

Nagy, G. 2004. "Classifiers that improve with use". In *Procs. Conference on Pattern Recognition and Multimedia*. (Tokyo, Japan 2004), p. 79–86. IEICE.

Narasimhamurthy, A. and L. I. Kuncheva. 2007. "A framework for generating data to simulate changing environments". In *Proc. of the 25th IASTED International Multi-Conf.: artificial intelligence and applications*. p. 84-389.

Nickabadi, A., M. M. Ebadzadeh, and R. Safabakhsh. 2008a. "DNPSO: a dynamic niching particle swarm optimizer for multi-modal optimization". In *Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on*. p. 26–32. IEEE.

Nickabadi, A., M. M. Ebadzadeh, and R. Safabakhsh. 2008b. "Evaluating the performance of dnpso in dynamic environments". In *2008 IEEE International Conference on Systems, Man and Cybernetics, SMC 2008, October 12, 2008 - October 15, 2008*. p. 2640-2645. Institute of Electrical and Electronics Engineers Inc.

Niinuma, N., U. Park, and A. Jain. 2010. "Soft Biometric Traits for Continuous User Authentication". *IEEE Trans. on Information Forensics and Security*, vol. 5, n° 4, p. 771–780.

Oh, I.-S. and C. Suen. 01 2002. "A class-modular feedforward neural network for handwriting recognition". *Pattern Recognition*, vol. 35, n° 1, p. 229–44.

Okamato, K., S. Ozawa, and S. Abe. 2003. "A fast incremental learning algorithm of RBF networks with long-term memory". In *2003 International Joint Conference on Neural Networks, 20-24 July 2003*. (Graduate Sch. of Sci. Technol., Kobe Univ., Japan 2003), p. 102–7. IEEE.

Ortíz Díaz, A., J. del Campo-Ávila, G. Ramos-Jiménez, I. Frías Blanco, Y. Caballero Mota, A. Mustelier Hechavarría, and R. Morales-Bueno. 2015. "Fast Adapting Ensemble: A New Algorithm for Mining Data Streams with Concept Drift". *The Scientific World Journal*, vol. 2015.

Oza, N. C. 2001. *Online ensemble learning*. Technical report.

Pagano, C., E. Granger, R. Sabourin, and D. O. Gorodnichy. 2012. "Detector ensembles for face recognition in video surveillance". In *Neural Networks (IJCNN), The 2012 International Joint Conference on*. p. 1–8. IEEE.

Pagano, C., E. Granger, R. Sabourin, G. Marcialis, and F. Roli. 2014. "Adaptive ensembles for face recognition in changing video surveillance environments". *Information Sciences*, vol. 286, p. 75–101.

Pagano, C., E. Granger, R. Sabourin, G. Marcialis, and F. Roli. 2015. "Adaptive Classification for Person Re-Identification Driven by Change Detection". In *I4th International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. (Lisbon, Portugal 2015).

Phillips, P. J., P. Grother, R. Micheals, D. M. Blackburn, E. Tabassi, and M. Bone. 2003. "Face recognition vendor test 2002". In *2003 IEEE International Workshop on Analysis and Modeling of Faces and Gestures, 17 Oct. 2003*. (DARPA, Arlington, VA, USA 2003). IEEE Comput. Soc.

Phillips, P. J., P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. 2005. "Overview of the Face Recognition Grand Challenge". In *Proc. of the 2005 Conference on Computer Vision and Pattern Recognition (CVPR'05)*. p. 947–954. IEEE.

Polikar, R., L. Upda, S. S. Upda, and V. Honavar. 2001. "Learn++: an incremental learning algorithm for supervised neural networks". *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 31, n° Copyright 2002, IEE, p. 497-508.

Proedrou, K., I. Nouretdinov, V. Vovk, and A. Gammerman. 2002. "Transductive confidence machines for pattern recognition". In *Machine Learning: ECML 2002. 13th European Conference on Machine Learning. Proceedings (Lecture Notes in Artificial Intelligence Vol.2430)*. (Berlin, Germany 2002), p. 381–90. Springer-Verlag.

Ramamurthy, S. and R. Bhatnagar. 2007. "Tracking recurrent concept drift in streaming data using ensemble classifiers". In *Machine Learning and Applications, 2007. ICMLA 2007. Sixth International Conference on*. p. 404–409. IEEE.

Rattani, A., B. Freni, G. L. Marcialis, and F. Roli. 2009. "Template update methods in adaptive biometric systems: A critical review". In *Lecture Notes in Computer Science (included Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. p. 847 - 856.

Rattani, A., G.-L. Marcialis, and F. Roli. 2013. "A multi-modal dataset, protocol and tools for adaptive biometric systems: a benchmarking study". *International Journal of Biometrics*, vol. 5, n° 4, p. 266 - 287.

Rattani, A. 2010. "Adaptive biometric system based on template update procedures". PhD thesis, PhD Thesis, University of Cagliari, Italy.

Rattani, A., G. L. Marcialis, and F. Roli. 2008. "Capturing large intra-class variations of biometric data by template co-updating". In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*. p. 1–6. IEEE.

Rattani, A., G. L. Marcialis, and F. Roli. 2011. "Self adaptive systems: An experimental analysis of the performance over time". In *Computational Intelligence in Biometrics and Identity Management (CIBIM), 2011 IEEE Workshop on*. p. 36–43. IEEE.

Riedmiller, M. 1994. "Advanced supervised learning in multi-layer perceptrons — From backpropagation to adaptive learning algorithms". *Computer Standards & Interfaces*, vol. 16, n° 3, p. 265 - 278.

Rokach, L. 2010. "Ensemble-based classifiers". *Artificial Intelligence Review*, vol. 33, n° 1-2, p. 1–39.

Roli, F. and G. L. Marcialis. 2006. Semi-supervised pca-based face recognition using self-training. *Structural, Syntactic, and Statistical Pattern Recognition*, p. 560–568. Springer.

Roli, F., L. Didaci, and G. Marcialis. 2008. Adaptive biometric systems that can improve with use. Ratha, N. and Venu Govindaraju, editors, *Advances in Biometrics*, p. 447-471. Springer London. ISBN 978-1-84628-920-0.

Ross, A. and A. Jain. 2003. "Information fusion in biometrics". *Pattern Recognition Letters*, vol. 24, n° Copyright 2004, IEE, p. 2115-25.

Ruping, S. 2001. "Incremental learning with support vector machines". (Los Alamitos, CA, USA 2001), p. 641 - 2.

Ryu, C., H. Kim, and A. K. Jain. 2006. "Template adaptation based fingerprint verification". In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*. p. 582–585. IEEE.

Santana, A., R. G. Soares, A. M. Canuto, and M. C. de Souto. 2006. "A dynamic classifier selection method to build ensembles using accuracy and diversity". In *Neural Networks, 2006. SBRN'06. Ninth Brazilian Symposium on*. p. 36–41. IEEE.

Sellahewa, H., S. Jassim, et al. 2010. "Image-quality-based adaptive face recognition". *Instrumentation and Measurement, IEEE Transactions on*, vol. 59, n° 4, p. 805–813.

Song, M. and H. Wang. 2005. "Highly efficient incremental estimation of gaussian mixture models for online data stream clustering". In *Defense and Security*. p. 174–183. International Society for Optics and Photonics.

Stallkamp, J., H. K. Ekenel, and R. Stiefelhagen. 2007. "Video-based face recognition on real-world data". In *2007 11th IEEE International Conference on Computer Vision, 14-21 Oct. 2007*. (Univ. of Karlsruhe, Karlsruhe, Germany 2007), p. 229–36. IEEE.

Tax, D. M. J. and R. P. W. Duin. 07 2008. "Growing a multi-class classifier with a reject option". *Pattern Recognition Letters*, vol. 29, n° 10, p. 1565–70.

Turk, M. A. and A. P. Pentland. 1991. "Face recognition using eigenfaces". In *Proceedings 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (91CH2983-5), 3-6 June 1991*. (Media Lab., MIT, Cambridge, MA, USA 1991), p. 586–91. IEEE Comput. Sco. Press.

Ulas, A., M. Semerci, O. T. Yildiz, and E. Alpaydin. 04 2009. "Incremental construction of classifier and discriminant ensembles". *Information Sciences*, vol. 179, n° 9, p. 1298–318.

Viola, P. and M. J. Jones. 2004. "Robust real-time face detection". *International Journal of Computer Vision*, vol. 57, p. 137-154.

Wang, Z. and A. C. Bovik. 2002. "A universal image quality index". *Signal Processing Letters, IEEE*, vol. 9, n° 3, p. 81–84.

Wang, Z., A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. "Image quality assessment: from error visibility to structural similarity". *IEEE Transactions on Image Processing*, vol. 13, n° 4, p. 600–612.

Weiss, G. M. 2003. "The effect of small disjuncts and class distribution on decision tree learning". PhD thesis, Rutgers, The State University of New Jersey.

Wong, Y., S. Chen, S. Mau, C. Sanderson, and B. C. Lovell. 2011. "Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition". In *Computer Vision and Pattern Recognition Workshops, 2011. CVPRW'11. IEEE Computer Society Conference on*. p. 74-81. IEEE.

Woods, K., K. Bowyer, and W. P. Kegelmeyer Jr. 1996. "Combination of multiple classifiers using local accuracy estimates". In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*. p. 391–396. IEEE.

Wright, J., A. Yang, A. Ganesh, and S. Sastry. 2009. "Robust Face Recognition via Sparse Representation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 2, p. 210–227.

Yang, M.-H., D. J. Kriegman, and N. Ahuja. 2002. "Detecting faces in images: A survey". *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, n° 1, p. 34–58.

Zhao, W., R. Chellappa, P. J. Phillips, and A. Rosenfeld. 12 2003. "Face recognition: a literature survey". *ACM Computing Surveys*, vol. 35, n° 4, p. 399–459.

Zhou, S., V. Krueger, and R. Chellappa. 07 2003. "Probabilistic recognition of human faces from video". *Computer Vision and Image Understanding*, vol. 91, n° 1-2, p. 214–45.

Zhou, S. K., R. Chellappa, and W. Zhao, 2006. *Unconstrained face recognition*, volume 5.

Zhu, X. 2005. "Semi-supervised learning literature survey".

Zhu, X., X. Wu, and Y. Yang. 2004. "Dynamic classifier selection for effective mining from noisy data streams". In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*. p. 305–312. IEEE.